An Introduction to Computational Fluid Dynamics

THE FINITE VOLUME METHOD

Second Edition

H K Versteeg and W Malalasekera

Harlow, England • London • New York • Boston • San Francisco • Toronto
Sydney • Tokyo • Singapore • Hong Kong • Seoul • Taipei • New Delhi
Cape Town • Madrid • Mexico City • Amsterdam • Munich • Paris • Milan

An Introduction to Computational Fluid Dynamics

Supporting resources

Visit **www.pearsoned.co.uk/versteeg** to find valuable online resources

For instructors

- PowerPoint slides that can be downloaded and used for presentations

For more information please contact your local Pearson Education sales representative or visit **www.pearsoned.co.uk/versteeg**

Contents

Preface

We were pleasantly surprised by the ready acceptance of the first edition of our book by the CFD community and by the amount of positive feedback received over a period of 10 years. To us this has provided justification of our original plan, which was to provide an accessible introduction to this fast-growing topic to support teaching at senior undergraduate level, postgraduate research and new industrial users of commercial CFD codes. Our second edition seeks to enhance and update. The structure and didactic approach of the first edition have been retained without change, but augmented by a selection of the most important developments in CFD.

In our treatment of the physics of fluid flows we have added a summary of the basic ideas underpinning large-eddy simulation (LES) and direct numerical simulation (DNS). These resource-intensive turbulence prediction techniques are likely to have a major impact in the medium term on CFD due to the increased availability of high-end computing capability.

Over the last decade a number of new discretisation techniques and solution approaches have come to the fore in commercial CFD codes. To reflect these developments we have included summaries of TVD techniques, which give stable, higher-order accurate solutions of convection–diffusion problems, and of iterative techniques and multi-grid accelerators that are now commonly used for the solution of systems of discretised equations. We have also added examples of the SIMPLE algorithm for pressure–velocity coupling to illustrate its workings.

At the time of writing our first edition, CFD was firmly established in the aerospace, automotive and power generation sectors. Subsequently, it has spread throughout engineering industry. This has gone hand in hand with major improvements in the treatment of complex geometries. We have devoted a new chapter to describing key aspects of unstructured meshing techniques that have made this possible.

Application of CFD results in industrial research and design crucially hinges on confidence in its outcomes. We have included a new chapter on uncertainty in CFD results. Given the rapid growth in CFD applications it is difficult to cover, within the space of a single introductory volume, even a small part of the submodelling methodology that is now included in many general-purpose CFD codes. Our selection of advanced application material covers combustion and radiation algorithms, which reflects our local perspective as mechanical engineers with interest in internal flow and combustion.

Finally, we thank colleagues in UK and overseas universities who have encouraged us with positive responses and constructive comments on our first edition and our proposals for a second edition. We are also grateful to several colleagues and postgraduate researchers who have given help in the

development of material, particularly Dr Jonathan Henson, Dr Mamdud Hossain, Dr Naminda Kandamby, Dr Andreas Haselbacher, Murthy Ravikanti-Veera and Anand Odedra.

August 2006 H.K. Versteeg
Loughborough W. Malalasekera

Acknowledgements

The authors wish to acknowledge the following persons, organisations and publishers for permission to reproduce from their publications in this book.

Professor H. Nagib for Figure 3.2, from Van Dyke, M. (1982) *An Album of Fluid Motion*, The Parabolic Press, Stanford; Professor S. Taneda and the Japan Society of Mechanical Engineers for Figure 3.7, from Nakayama, Y. (1988) *Visualised Flow*, compiled by the Japan Society of Mechanical Engineers, Pergamon Press; Professor W. Fiszdon and the Polish Academy of Sciences for Figure 3.9, from Van Dyke, M. (1982) *An Album of Fluid Motion*, The Parabolic Press, Stanford; Figures 3.11 and 3.14 from Schlichting, H. (1979) *Boundary Layer Theory*, 7th edn, reproduced with permission of The McGraw-Hill Companies; Figure 3.15 reprinted by permission of Elsevier Science from 'Calculation of Turbulent Reacting Flows: A Review' by W. P. Jones and J. H. Whitelaw, *Combustion and Flame*, Vol. 48, pp. 1–26, © 1982 by The Combustion Institute; Figure 3.16 from Leschziner, M. A. (2000) 'The Computation of Turbulent Engineering Flows', in R. Peyret and E. Krause (eds) *Advanced Turbulent Flow Computations*, reproduced with permission from Springer Wien NewYork; Figures 3.17 and 3.18 reprinted from *International Journal of Heat and Fluid Flow*, Vol. 23, Moin, P., 'Advances in Large Eddy Simulation Methodology for Complex Flows', pp. 710–712, © 2002, with permission from Elsevier; Dr Andreas Haselbacher for Figures 11.2, 11.9 and 11.11, from 'A Grid-Transparent Numerical Method for Compressible Viscous Flows on Mixed Unstructured Grids', thesis, Loughborough University; The Combustion Institute for Figure 12.8, from Magnussen, B. F. and Hjertager, B. H. (1976) 'On the Mathematical Modelling of Turbulent Combustion with Special Emphasis on Soot Formation and Combustion', Sixteenth Symposium (Int.) on Combustion, and Figure 12.9, from Gosman, A. D., Lockwood, F. C. and Salooja, A. P. (1978) 'The Prediction of Cylindrical Furnaces Gaseous Fuelled with Premixed and Diffusion Burners', Seventeenth Symposium (Int.) on Combustion; Gordon and Breach Science Publishers for Figure 12.10, from Nikjooy, M., So, R. M. C. and Peck, R. E. (1988) 'Modelling of Jet- and Swirl-stabilised Reacting Flows in Axisymmetric Combustors', *Combust. Sci and Tech.* © 1988 by Gordon and Breach Science Publishers.

In some instances we have been unable to trace the owners of copyright material, and we would appreciate any information that would enable us to do so.

Chapter one Introduction

1.1 What is CFD?

Computational fluid dynamics or CFD is the analysis of systems involving fluid flow, heat transfer and associated phenomena such as chemical reactions by means of computer-based simulation. The technique is very powerful and spans a wide range of industrial and non-industrial application areas. Some examples are:

- aerodynamics of aircraft and vehicles: lift and drag
- hydrodynamics of ships
- power plant: combustion in internal combustion engines and gas turbines
- turbomachinery: flows inside rotating passages, diffusers etc.
- electrical and electronic engineering: cooling of equipment including microcircuits
- chemical process engineering: mixing and separation, polymer moulding
- external and internal environment of buildings: wind loading and heating/ventilation
- marine engineering: loads on off-shore structures
- environmental engineering: distribution of pollutants and effluents
- hydrology and oceanography: flows in rivers, estuaries, oceans
- meteorology: weather prediction
- biomedical engineering: blood flows through arteries and veins

From the 1960s onwards the aerospace industry has integrated CFD techniques into the design, R&D and manufacture of aircraft and jet engines. More recently the methods have been applied to the design of internal combustion engines, combustion chambers of gas turbines and furnaces. Furthermore, motor vehicle manufacturers now routinely predict drag forces, under-bonnet air flows and the in-car environment with CFD. Increasingly CFD is becoming a vital component in the design of industrial products and processes.

The ultimate aim of developments in the CFD field is to provide a capability comparable with other CAE (computer-aided engineering) tools such as stress analysis codes. The main reason why CFD has lagged behind is the tremendous complexity of the underlying behaviour, which precludes a description of fluid flows that is at the same time economical and sufficiently complete. The availability of affordable high-performance computing hardware and the introduction of user-friendly interfaces have led to a recent upsurge of interest, and CFD has entered into the wider industrial community since the 1990s.

We estimate the minimum cost of suitable hardware to be between £5,000 and £10,000 (plus annual maintenance costs). The perpetual licence fee for commercial software typically ranges from £10,000 to £50,000 depending on the number of 'added extras' required. CFD software houses can usually arrange annual licences as an alternative. Clearly the investment costs of a CFD capability are not small, but the total expense is not normally as great as that of a high-quality experimental facility. Moreover, there are several unique advantages of CFD over experiment-based approaches to fluid systems design:

- substantial reduction of lead times and costs of new designs
- ability to study systems where controlled experiments are difficult or impossible to perform (e.g. very large systems)
- ability to study systems under hazardous conditions at and beyond their normal performance limits (e.g. safety studies and accident scenarios)
- practically unlimited level of detail of results

The variable cost of an experiment, in terms of facility hire and/or person-hour costs, is proportional to the number of data points and the number of configurations tested. In contrast, CFD codes can produce extremely large volumes of results at virtually no added expense, and it is very cheap to perform parametric studies, for instance to optimise equipment performance.

Below we look at the overall structure of a CFD code and discuss the role of the individual building blocks. We also note that, in addition to a substantial investment outlay, an organisation needs qualified people to run the codes and communicate their results, and briefly consider the modelling skills required by CFD users. We complete this otherwise upbeat section by wondering whether the next constraint to the further spread of CFD amongst the industrial community could be a scarcity of suitably trained personnel instead of availability and/or cost of hardware and software.

1.2 How does a CFD code work?

CFD codes are structured around the numerical algorithms that can tackle fluid flow problems. In order to provide easy access to their solving power all commercial CFD packages include sophisticated user interfaces to input problem parameters and to examine the results. Hence all codes contain three main elements: (i) a pre-processor, (ii) a solver and (iii) a post-processor. We briefly examine the function of each of these elements within the context of a CFD code.

Pre-processor

Pre-processing consists of the input of a flow problem to a CFD program by means of an operator-friendly interface and the subsequent transformation of this input into a form suitable for use by the solver. The user activities at the pre-processing stage involve:

- Definition of the geometry of the region of interest: the computational **domain**
- Grid generation – the sub-division of the domain into a number of smaller, non-overlapping sub-domains: a **grid** (or **mesh**) of **cells** (or **control volumes** or elements)
- Selection of the physical and chemical phenomena that need to be modelled

- Definition of fluid properties
- Specification of appropriate boundary conditions at cells which coincide with or touch the domain boundary

The solution to a flow problem (velocity, pressure, temperature etc.) is defined at **nodes** inside each cell. The accuracy of a CFD solution is governed by the number of cells in the grid. In general, the larger the number of cells, the better the solution accuracy. Both the accuracy of a solution and its cost in terms of necessary computer hardware and calculation time are dependent on the fineness of the grid. Optimal meshes are often non-uniform: finer in areas where large variations occur from point to point and coarser in regions with relatively little change. Efforts are under way to develop CFD codes with a (self-)adaptive meshing capability. Ultimately such programs will automatically refine the grid in areas of rapid variations. A substantial amount of basic development work still needs to be done before these techniques are robust enough to be incorporated into commercial CFD codes. At present it is still up to the skills of the CFD user to design a grid that is a suitable compromise between desired accuracy and solution cost.

Over 50% of the time spent in industry on a CFD project is devoted to the definition of the domain geometry and grid generation. In order to maximise productivity of CFD personnel all the major codes now include their own CAD-style interface and/or facilities to import data from proprietary surface modellers and mesh generators such as PATRAN and I-DEAS. Up-to-date pre-processors also give the user access to libraries of material properties for common fluids and a facility to invoke special physical and chemical process models (e.g. turbulence models, radiative heat transfer, combustion models) alongside the main fluid flow equations.

Solver

There are three distinct streams of numerical solution techniques: finite difference, finite element and spectral methods. We shall be solely concerned with the finite volume method, a special finite difference formulation that is central to the most well-established CFD codes: CFX/ANSYS, FLUENT, PHOENICS and STAR-CD. In outline the numerical algorithm consists of the following steps:

- Integration of the governing equations of fluid flow over all the (finite) control volumes of the domain
- Discretisation – conversion of the resulting integral equations into a system of algebraic equations
- Solution of the algebraic equations by an iterative method

The first step, the control volume integration, distinguishes the finite volume method from all other CFD techniques. The resulting statements express the (exact) conservation of relevant properties for each finite size cell. This clear relationship between the numerical algorithm and the underlying physical conservation principle forms one of the main attractions of the finite volume method and makes its concepts much more simple to understand by engineers than the finite element and spectral methods. The conservation of a general flow variable ϕ, e.g. a velocity component or enthalpy, within a finite control volume can be expressed as a balance between the various processes tending to increase or decrease it. In words we have:

[Rate of change of ϕ in the control volume with respect to time] = [Net rate of increase of ϕ due to convection into the control volume] + [Net rate of increase of ϕ due to diffusion into the control volume] + [Net rate of creation of ϕ inside the control volume]

CFD codes contain discretisation techniques suitable for the treatment of the key transport phenomena, convection (transport due to fluid flow) and diffusion (transport due to variations of ϕ from point to point) as well as for the source terms (associated with the creation or destruction of ϕ) and the rate of change with respect to time. The underlying physical phenomena are complex and non-linear so an iterative solution approach is required. The most popular solution procedures are by the TDMA (tri-diagonal matrix algorithm) line-by-line solver of the algebraic equations and the SIMPLE algorithm to ensure correct linkage between pressure and velocity. Commercial codes may also give the user a selection of further, more recent, techniques such as Gauss–Seidel point iterative techniques with multigrid accelerators and conjugate gradient methods.

Post-processor

As in pre-processing, a huge amount of development work has recently taken place in the post-processing field. Due to the increased popularity of engineering workstations, many of which have outstanding graphics capabilities, the leading CFD packages are now equipped with versatile data visualisation tools. These include:

- Domain geometry and grid display
- Vector plots
- Line and shaded contour plots
- 2D and 3D surface plots
- Particle tracking
- View manipulation (translation, rotation, scaling etc.)
- Colour PostScript output

More recently these facilities may also include animation for dynamic result display, and in addition to graphics all codes produce trusty alphanumeric output and have data export facilities for further manipulation external to the code. As in many other branches of CAE, the graphics output capabilities of CFD codes have revolutionised the communication of ideas to the non-specialist.

1.3 Problem solving with CFD

In solving fluid flow problems we need to be aware that the underlying physics is complex and the results generated by a CFD code are at best as good as the physics (and chemistry) embedded in it and at worst as good as its operator. Elaborating on the latter issue first, the user of a code must have skills in a number of areas. Prior to setting up and running a CFD simulation there is a stage of identification and formulation of the flow problem in terms of the physical and chemical phenomena that need to be considered. Typical decisions that might be needed are whether to model a problem in two or three dimensions, to exclude the effects of ambient temperature

or pressure variations on the density of an air flow, to choose to solve the turbulent flow equations or to neglect the effects of small air bubbles dissolved in tap water. To make the right choices requires good modelling skills, because in all but the simplest problems we need to make assumptions to reduce the complexity to a manageable level whilst preserving the salient features of the problem at hand. It is the appropriateness of the simplifications introduced at this stage that at least partly governs the quality of the information generated by CFD, so the user must continually be aware of all the assumptions, clear-cut and tacit ones, that have been made.

Performing the computation itself requires operator skills of a different kind. Specification of the domain geometry and grid design are the main tasks at the input stage and subsequently the user needs to obtain a successful simulation result. The two aspects that characterise such a result are convergence and grid independence. The solution algorithm is iterative in nature, and in a converged solution the so-called residuals – measures of the overall conservation of the flow properties – are very small. Progress towards a converged solution can be greatly assisted by careful selection of the settings of various relaxation factors and acceleration devices. There are no straightforward guidelines for making these choices since they are problem dependent. Optimisation of the solution speed requires considerable experience with the code itself, which can only be acquired by extensive use. There is no formal way of estimating the errors introduced by inadequate grid design for a general flow. Good initial grid design relies largely on an insight into the expected properties of the flow. A background in the fluid dynamics of the particular problem certainly helps, and experience with gridding of similar problems is also invaluable. The only way to eliminate errors due to coarseness of a grid is to perform a grid dependence study, which is a procedure of successive refinement of an initially coarse grid until certain key results do not change. Then the simulation is grid independent. A systematic search for grid-independent results forms an essential part of all high-quality CFD studies.

Every numerical algorithm has its own characteristic error patterns. Well-known CFD euphemisms for the word 'error' are terms such as numerical diffusion, false diffusion or even numerical flow. The likely error patterns can only be guessed on the basis of a thorough knowledge of the algorithms. At the end of a simulation the user must make a judgement whether the results are 'good enough'. It is impossible to assess the validity of the models of physics and chemistry embedded in a program as complex as a CFD code or the accuracy of its final results by any means other than comparison with experimental test work. Anyone wishing to use CFD in a serious way must realise that it is no substitute for experimentation, but a very powerful additional problem solving tool. Validation of a CFD code requires highly detailed information concerning the boundary conditions of a problem, and generates a large volume of results. To validate these in a meaningful way it is necessary to produce experimental data of similar scope. This may involve a programme of flow velocity measurements with hot-wire anemometry, laser Doppler anemometry or particle image velocimetry. However, if the environment is too hostile for such delicate laboratory equipment or if it is simply not available, static pressure and temperature measurements complemented by pitot-static tube traverses can also be useful to validate some aspects of a flow field.

Sometimes the facilities to perform experimental work may not (yet) exist, in which case the CFD user must rely on (i) previous experience,

(ii) comparisons with analytical solutions of similar but simpler flows and (iii) comparisons with high-quality data from closely related problems reported in the literature. Excellent sources of the last type of information can be found in *Transactions of the ASME* (in particular the *Journal of Fluids Engineering*, *Journal of Engineering for Gas Turbines and Power* and *Journal of Heat Transfer*), *AIAA Journal*, *Journal of Fluid Mechanics* and *Proceedings of the IMechE*.

CFD computation involves the creation of a set of numbers that (hopefully) constitutes a realistic approximation of a real-life system. One of the advantages of CFD is that the user has an almost unlimited choice of the level of detail of the results, but in the prescient words of C. Hastings, written in the pre-IT days of 1955: 'The purpose of computing is **insight** not numbers.' The underlying message is rightly cautionary. We should make sure that the main outcome of any CFD exercise is improved understanding of the behaviour of a system, but since there are no cast-iron guarantees with regard to the accuracy of a simulation, we need to validate our results frequently and stringently.

It is clear that there are guidelines for good operating practice which can assist the user of a CFD code, and repeated validation plays a key role as the final quality control mechanism. However, the main ingredients for success in CFD are experience and a thorough understanding of the physics of fluid flows and the fundamentals of the numerical algorithms. Without these it is very unlikely that the user will get the best out of a code. It is the intention of this book to provide all the necessary background material for a good understanding of the internal workings of a CFD code and its successful operation.

1.4 Scope of this book

This book seeks to present all the fundamental material needed for good simulation of fluid flows by means of the finite volume method, and is split into three parts. The first part, consisting of Chapters 2 and 3, is concerned with the fundamentals of fluid flows in three dimensions and turbulence. The treatment starts with the derivation of the governing partial differential equations of fluid flows in Cartesian co-ordinates. We stress the commonalities in the resulting conservation equations and arrive at the so-called transport equation, which is the basic form for the development of the numerical algorithms that are to follow. Moreover, we look at the auxiliary conditions required to specify a well-posed problem from a general perspective and quote a set of recommended boundary conditions and a number of derived ones that are useful in CFD practice. Chapter 3 represents the development of the concepts of turbulence that are necessary for a full appreciation of the finer details of CFD in many engineering applications. We look at the physics of turbulence and the characteristics of some simple turbulent flows and at the consequences of the appearance of random fluctuations on the flow equations. The resulting equations are not a closed or solvable set unless we introduce a turbulence model. We discuss the principal turbulence models that are used in industrial CFD, focusing our attention on the k–ε model, which is very popular in general-purpose flow computations. Some of the more recent developments that are likely to have a major impact on CFD in the near future are also reviewed.

Readers who are already familiar with the derivation of the 3D flow equations can move on to the second part without loss of continuity. Apart from

the discussion of the k–ε turbulence model, to which we return later, the material in Chapters 2 and 3 is largely self-contained. This allows the use of this book by those wishing to concentrate principally on the numerical algorithms, but requiring an overview of the fluid dynamics and the mathematics behind it for occasional reference in the same text.

The second part of the book is devoted to the numerical algorithms of the finite volume method and covers Chapters 4 to 9. Discretisation schemes and solution procedures for steady flows are discussed in Chapters 4 to 7. Chapter 4 describes the basic approach and derives the central difference scheme for diffusion phenomena. In Chapter 5 we emphasise the key properties of discretisation schemes, conservativeness, boundedness and transportiveness, which are used as a basis for the further development of the upwind, hybrid, QUICK and TVD schemes for the discretisation of convective terms. The non-linear nature of the underlying flow phenomena and the linkage between pressure and velocity in variable density fluid flows requires special treatment, which is the subject of Chapter 6. We introduce the SIMPLE algorithm and some of its more recent derivatives and also discuss the PISO algorithm. In Chapter 7 we describe algorithms for the solution of the systems of algebraic equations that appear after the discretisation stage. We focus our attention on the well-known TDMA algorithm, which was the basis of early CFD codes, and point iterative methods with multigrid accelerators, which are the current solvers of choice.

The theory behind all the numerical methods is developed around a set of worked examples which can be easily programmed on a PC. This presentation gives the opportunity for a detailed examination of all aspects of the discretisation schemes, which form the basic building blocks of practical CFD codes, including the characteristics of their solutions.

In Chapter 8 we assess the advantages and limitations of various schemes to deal with unsteady flows, and Chapter 9 completes the development of the numerical algorithms by considering the practical implementation of the most common boundary conditions in the finite volume method.

The book is primarily aimed to support those who have access to a CFD package, so that the issues raised in the text can be explored in greater depth. The solution procedures are nevertheless sufficiently well documented for the interested reader to be able to start developing a CFD code from scratch.

The third part of the book consists of a selection of topics relating to the application of the finite volume method to complex industrial problems. In Chapter 10 we review aspects of accuracy and uncertainty in CFD. It is not possible to predict the error in a CFD result from first principles, which creates some problems for the industrial user who wishes to evolve equipment design on the basis of insights gleaned from CFD. In order to address this issue a systematic process has been developed to assist in the quantification of the uncertainty of CFD output. We discuss methods, the concepts of verification and validation, and give a summary of rules for best practice that have been developed by the CFD community to assist users. In Chapter 11 we discuss techniques to cope with complex geometry. We review various approaches based on structured meshes: Cartesian co-ordinate systems, generalised co-ordinate systems based on transformations, and block-structured grids, which enable design of specific meshes tailored to the needs of different parts of geometry. We give details of the implementation of the finite volume method on unstructured meshes. These are not based on a grid of lines to define nodal positions and can include control volumes that can have

any shape. Consequently, unstructured meshes have the ability to match the boundary shape of CFD solution domains of arbitrary complexity. This greatly facilitates the design and refinement of meshes, so that unstructured meshes are the most popular method in industrial CFD applications. The remaining Chapters 12 and 13 are concerned with one of the most significant engineering applications of CFD: energy technology and combusting systems. In order to provide a self-contained introduction to the most important aspects of CFD in reacting flows, we introduce in Chapter 12 the basic thermodynamic and chemical concepts of combustion and review the most important models of combustion. Our particular focus is the laminar flamelet model of non-premixed turbulent combustion, which is the most widely researched model with capabilities to predict the main combustion reaction *and* pollutant species concentrations. In the final Chapter 13 we discuss CFD techniques to predict radiative heat transfer, a good understanding of which is necessary for accurate combustion calculations.

Chapter two Conservation laws of fluid motion and boundary conditions

In this chapter we develop the mathematical basis for a comprehensive general-purpose model of fluid flow and heat transfer from the basic principles of conservation of mass, momentum and energy. This leads to the governing equations of fluid flow and a discussion of the necessary auxiliary conditions – initial and boundary conditions. The main issues covered in this context are:

- Derivation of the system of partial differential equations (PDEs) that govern flows in Cartesian (x, y, z) co-ordinates
- Thermodynamic equations of state
- Newtonian model of viscous stresses leading to the Navier–Stokes equations
- Commonalities between the governing PDEs and the definition of the transport equation
- Integrated forms of the transport equation over a finite time interval and a finite control volume
- Classification of physical behaviours into three categories: elliptic, parabolic and hyperbolic
- Appropriate boundary conditions for each category
- Classification of fluid flows
- Auxiliary conditions for viscous fluid flows
- Problems with boundary condition specification in high Reynolds number and high Mach number flows

2.1 Governing equations of fluid flow and heat transfer

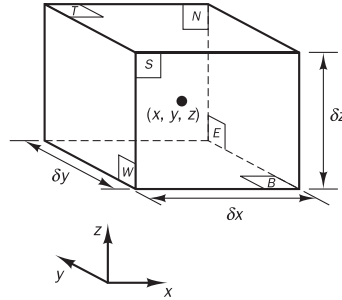
The governing equations of fluid flow represent mathematical statements of the **conservation laws of physics**:

- The mass of a fluid is conserved
- The rate of change of momentum equals the sum of the forces on a fluid particle (Newton's second law)
- The rate of change of energy is equal to the sum of the rate of heat addition to and the rate of work done on a fluid particle (first law of thermodynamics)

The fluid will be regarded as a continuum. For the analysis of fluid flows at macroscopic length scales (say 1 μm and larger) the molecular structure of matter and molecular motions may be ignored. We describe the behaviour of the fluid in terms of macroscopic properties, such as velocity, pressure, density and temperature, and their space and time derivatives. These may

be thought of as averages over suitably large numbers of molecules. A fluid particle or point in a fluid is then the smallest possible element of fluid whose macroscopic properties are not influenced by individual molecules.

We consider such a small element of fluid with sides δx, δy and δz (Figure 2.1).

Figure 2.1 Fluid element for conservation laws

The six faces are labelled N, S, E, W, T and B, which stands for North, South, East, West, Top and Bottom. The positive directions along the co-ordinate axes are also given. The centre of the element is located at position (x, y, z). A systematic account of changes in the mass, momentum and energy of the fluid element due to fluid flow across its boundaries and, where appropriate, due to the action of sources inside the element, leads to the fluid flow equations.

All fluid properties are functions of space and time so we would strictly need to write $\rho(x, y, z, t)$, $p(x, y, z, t)$, $T(x, y, z, t)$ and $\mathbf{u}(x, y, z, t)$ for the density, pressure, temperature and the velocity vector respectively. To avoid unduly cumbersome notation we will not explicitly state the dependence on space co-ordinates and time. For instance, the density at the centre (x, y, z) of a fluid element at time t is denoted by ρ and the x-derivative of, say, pressure p at (x, y, z) and time t by $\partial p/\partial x$. This practice will also be followed for all other fluid properties.

The element under consideration is so small that fluid properties at the faces can be expressed accurately enough by means of the first two terms of a Taylor series expansion. So, for example, the pressure at the W and E faces, which are both at a distance of $\frac{1}{2}\delta x$ from the element centre, can be expressed as

$$p - \frac{\partial p}{\partial x}\frac{1}{2}\delta x \quad \text{and} \quad p + \frac{\partial p}{\partial x}\frac{1}{2}\delta x$$

2.1.1 Mass conservation in three dimensions

The first step in the derivation of the mass conservation equation is to write down a mass balance for the fluid element:

Rate of increase of mass in fluid element	=	Net rate of flow of mass into fluid element

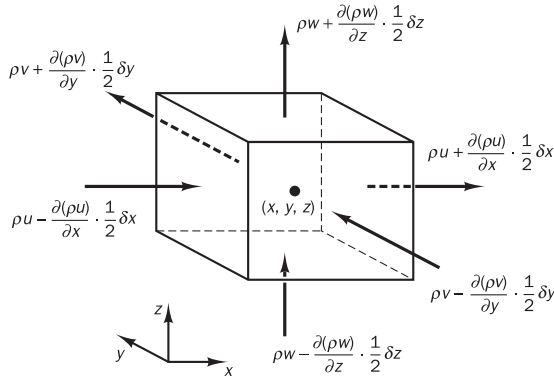
The rate of increase of mass in the fluid element is

$$\frac{\partial}{\partial t}(\rho \delta x \delta y \delta z) = \frac{\partial \rho}{\partial t}\delta x \delta y \delta z \tag{2.1}$$

Next we need to account for the mass flow rate across a face of the element, which is given by the product of density, area and the velocity component normal to the face. From Figure 2.2 it can be seen that the net rate of flow of mass into the element across its boundaries is given by

$$\left(\rho u - \frac{\partial(\rho u)}{\partial x}\frac{1}{2}\delta x\right)\delta y \delta z - \left(\rho u + \frac{\partial(\rho u)}{\partial x}\frac{1}{2}\delta x\right)\delta y \delta z$$
$$+ \left(\rho v - \frac{\partial(\rho v)}{\partial y}\frac{1}{2}\delta y\right)\delta x \delta z - \left(\rho v + \frac{\partial(\rho v)}{\partial y}\frac{1}{2}\delta y\right)\delta x \delta z$$
$$+ \left(\rho w - \frac{\partial(\rho w)}{\partial z}\frac{1}{2}\delta z\right)\delta x \delta y - \left(\rho w + \frac{\partial(\rho w)}{\partial z}\frac{1}{2}\delta z\right)\delta x \delta y \tag{2.2}$$

Flows which are directed into the element produce an increase of mass in the element and get a positive sign and those flows that are leaving the element are given a negative sign.

Figure 2.2 Mass flows in and out of fluid element

The rate of increase of mass inside the element (2.1) is now equated to the net rate of flow of mass into the element across its faces (2.2). All terms of the resulting mass balance are arranged on the left hand side of the equals sign and the expression is divided by the element volume $\delta x \delta y \delta z$. This yields

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \tag{2.3}$$

or in more compact vector notation

$$\boxed{\frac{\partial \rho}{\partial t} + \text{div}(\rho \mathbf{u}) = 0} \tag{2.4}$$

Equation (2.4) is the **unsteady, three-dimensional mass conservation or continuity equation** at a point in a **compressible fluid**. The first term

on the left hand side is the rate of change in time of the density (mass per unit volume). The second term describes the net flow of mass out of the element across its boundaries and is called the convective term.

For an **incompressible fluid** (i.e. a liquid) the density ρ is constant and equation (2.4) becomes

$$\text{div } \mathbf{u} = 0 \tag{2.5}$$

or in longhand notation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.6}$$

2.1.2 Rates of change following a fluid particle and for a fluid element

The momentum and energy conservation laws make statements regarding changes of properties of a fluid particle. This is termed the Lagrangian approach. Each property of such a particle is a function of the position (x, y, z) of the particle and time t. Let the value of a property per unit mass be denoted by ϕ. The total or substantive derivative of ϕ with respect to time following a fluid particle, written as $D\phi/Dt$, is

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial x}\frac{dx}{dt} + \frac{\partial \phi}{\partial y}\frac{dy}{dt} + \frac{\partial \phi}{\partial z}\frac{dz}{dt}$$

A fluid particle follows the flow, so $dx/dt = u$, $dy/dt = v$ and $dz/dt = w$. Hence the substantive derivative of ϕ is given by

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} + v\frac{\partial \phi}{\partial y} + w\frac{\partial \phi}{\partial z} = \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \text{grad } \phi \tag{2.7}$$

$D\phi/Dt$ defines rate of change of property ϕ per unit mass. It is possible to develop numerical methods for fluid flow calculations based on the Lagrangian approach, i.e. by tracking the motion and computing the rates of change of conserved properties ϕ for collections of fluid particles. However, it is far more common to develop equations for collections of fluid elements making up a region fixed in space, for example a region defined by a duct, a pump, a furnace or similar piece of engineering equipment. This is termed the Eulerian approach.

As in the case of the mass conservation equation, we are interested in developing equations for rates of change per unit volume. The rate of change of property ϕ per unit volume for a fluid particle is given by the product of $D\phi/Dt$ and density ρ, hence

$$\rho\frac{D\phi}{Dt} = \rho\left(\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \text{grad } \phi\right) \tag{2.8}$$

The most useful forms of the conservation laws for fluid flow computation are concerned with changes of a flow property for a fluid element that is stationary in space. The relationship between the substantive derivative of ϕ, which follows a fluid particle, and rate of change of ϕ for a fluid element is now developed.

The mass conservation equation contains the mass per unit volume (i.e. the density ρ) as the conserved quantity. The sum of the rate of change of density in time and the convective term in the mass conservation equation (2.4) for a fluid element is

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u})$$

The generalisation of these terms for an arbitrary conserved property is

$$\frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) \tag{2.9}$$

Formula (2.9) expresses the rate of change in time of ϕ per unit volume plus the net flow of ϕ out of the fluid element per unit volume. It is now rewritten to illustrate its relationship with the substantive derivative of ϕ:

$$\frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) = \rho\left[\frac{\partial \phi}{\partial t} + \mathbf{u} \,.\, \operatorname{grad} \phi\right] + \phi\left[\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho\mathbf{u})\right]$$
$$= \rho\frac{D\phi}{Dt} \tag{2.10}$$

The term $\phi[(\partial\rho/\partial t) + \operatorname{div}(\rho\mathbf{u})]$ is equal to zero by virtue of mass conservation (2.4). In words, relationship (2.10) states

Rate of increase of ϕ of fluid element	+	Net rate of flow of ϕ out of fluid element	=	Rate of increase of ϕ for a fluid particle

To construct the three components of the momentum equation and the energy equation the relevant entries for ϕ and their rates of change per unit volume as defined in (2.8) and (2.10) are given below:

x-momentum	u	$\rho\frac{Du}{Dt}$	$\frac{\partial(\rho u)}{\partial t} + \operatorname{div}(\rho u\mathbf{u})$
y-momentum	v	$\rho\frac{Dv}{Dt}$	$\frac{\partial(\rho v)}{\partial t} + \operatorname{div}(\rho v\mathbf{u})$
z-momentum	w	$\rho\frac{Dw}{Dt}$	$\frac{\partial(\rho w)}{\partial t} + \operatorname{div}(\rho w\mathbf{u})$
energy	E	$\rho\frac{DE}{Dt}$	$\frac{\partial(\rho E)}{\partial t} + \operatorname{div}(\rho E\mathbf{u})$

Both the conservative (or divergence) form and non-conservative form of the rate of change can be used as alternatives to express the conservation of a physical quantity. The non-conservative forms are used in the derivations of momentum and energy equations for a fluid flow in sections 2.4 and 2.5 for brevity of notation and to emphasise that the conservation laws are fundamentally conceived as statements that apply to a particle of fluid. In the final

section 2.8 we will return to the conservative form that is used in finite volume CFD calculations.

2.1.3 Momentum equation in three dimensions

Newton's second law states that the rate of change of momentum of a fluid particle equals the sum of the forces on the particle:

Rate of increase of momentum of fluid particle	=	Sum of forces on fluid particle

The **rates of increase of *x*-, *y*- and *z*-momentum** per unit volume of a fluid particle are given by

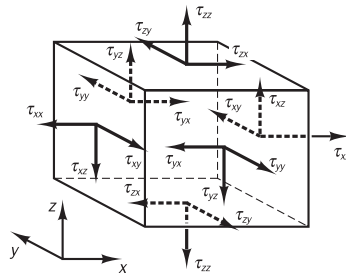
$$\rho\frac{Du}{Dt} \qquad \rho\frac{Dv}{Dt} \qquad \rho\frac{Dw}{Dt} \tag{2.11}$$

We distinguish two types of **forces** on fluid particles:

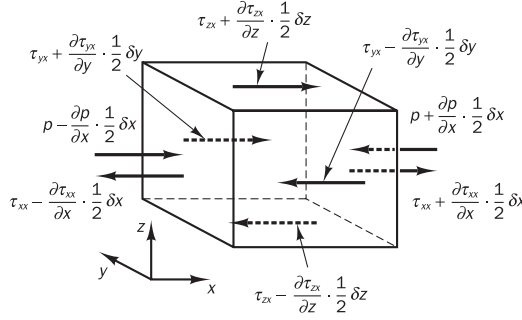
- **surface forces**
  - pressure forces
  - viscous forces
  - gravity force
- **body forces**
  - centrifugal force
  - Coriolis force
  - electromagnetic force

It is common practice to highlight the contributions due to the surface forces as separate terms in the momentum equation and to include the effects of body forces as source terms.

The state of stress of a fluid element is defined in terms of the pressure and the nine viscous stress components shown in Figure 2.3. The pressure, a normal stress, is denoted by p. Viscous stresses are denoted by τ. The usual suffix notation τ_{ij} is applied to indicate the direction of the viscous stresses. The suffices i and j in τ_{ij} indicate that the stress component acts in the j-direction on a surface normal to the i-direction.

Figure 2.3 Stress components on three faces of fluid element

First we consider the x-components of the forces due to pressure p and stress components τ_{xx}, τ_{yx} and τ_{zx} shown in Figure 2.4. The magnitude of a

force resulting from a surface stress is the product of stress and area. Forces aligned with the direction of a co-ordinate axis get a positive sign and those in the opposite direction a negative sign. The net force in the x-direction is the sum of the force components acting in that direction on the fluid element.

Figure 2.4 Stress components in the x-direction

On the pair of faces (E, W) we have

$$\left[\left(p - \frac{\partial p}{\partial x}\frac{1}{2}\delta x\right) - \left(\tau_{xx} - \frac{\partial \tau_{xx}}{\partial x}\frac{1}{2}\delta x\right)\right]\delta y \delta z + \left[-\left(p + \frac{\partial p}{\partial x}\frac{1}{2}\delta x\right) + \left(\tau_{xx} + \frac{\partial \tau_{xx}}{\partial x}\frac{1}{2}\delta x\right)\right]\delta y \delta z = \left(-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x}\right)\delta x \delta y \delta z \quad (2.12a)$$

The net force in the x-direction on the pair of faces (N, S) is

$$-\left(\tau_{yx} - \frac{\partial \tau_{yx}}{\partial y}\frac{1}{2}\delta y\right)\delta x \delta z + \left(\tau_{yx} + \frac{\partial \tau_{yx}}{\partial y}\frac{1}{2}\delta y\right)\delta x \delta z = \frac{\partial \tau_{yx}}{\partial y}\delta x \delta y \delta z \quad (2.12b)$$

Finally the net force in the x-direction on faces T and B is given by

$$-\left(\tau_{zx} - \frac{\partial \tau_{zx}}{\partial z}\frac{1}{2}\delta z\right)\delta x \delta y + \left(\tau_{zx} + \frac{\partial \tau_{zx}}{\partial z}\frac{1}{2}\delta z\right)\delta x \delta y = \frac{\partial \tau_{zx}}{\partial z}\delta x \delta y \delta z \quad (2.12c)$$

The total force per unit volume on the fluid due to these surface stresses is equal to the sum of (2.12a), (2.12b) and (2.12c) divided by the volume $\delta x \delta y \delta z$:

$$\frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} \quad (2.13)$$

Without considering the body forces in further detail their overall effect can be included by defining a source S_{Mx} of x-momentum per unit volume per unit time.

The ***x*-component of the momentum equation** is found by setting the rate of change of x-momentum of the fluid particle (2.11) equal to the

total force in the x-direction on the element due to surface stresses (2.13) plus the rate of increase of x-momentum due to sources:

$$\rho \frac{Du}{Dt} = \frac{\partial(-p + \tau_{xx})}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + S_{Mx} \tag{2.14a}$$

It is not too difficult to verify that the ***y*-component of the momentum equation** is given by

$$\rho \frac{Dv}{Dt} = \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial(-p + \tau_{yy})}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + S_{My} \tag{2.14b}$$

and the ***z*-component of the momentum equation** by

$$\rho \frac{Dw}{Dt} = \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial(-p + \tau_{zz})}{\partial z} + S_{Mz} \tag{2.14c}$$

The sign associated with the pressure is opposite to that associated with the normal viscous stress, because the usual sign convention takes a tensile stress to be the positive normal stress so that the pressure, which is by definition a compressive normal stress, has a minus sign.

The effects of surface stresses are accounted for explicitly; the source terms S_{Mx}, S_{My} and S_{Mz} in (2.14a–c) include contributions due to body forces only. For example, the body force due to gravity would be modelled by $S_{Mx} = 0$, $S_{My} = 0$ and $S_{Mz} = -\rho g$.

2.1.4 Energy equation in three dimensions

The energy equation is derived from the **first law of thermodynamics**, which states that the rate of change of energy of a fluid particle is equal to the rate of heat addition to the fluid particle plus the rate of work done on the particle:

Rate of increase of energy of fluid particle	=	Net rate of heat added to fluid particle	+	Net rate of work done on fluid particle

As before, we will be deriving an equation for the **rate of increase of energy** of a fluid particle per unit volume, which is given by

$$\rho \frac{DE}{Dt} \tag{2.15}$$

Work done by surface forces

The **rate of work done** on the fluid particle in the element by a **surface force** is equal to the product of the force and velocity component in the direction of the force. For example, the forces given by (2.12a–c) all act in the x-direction. The work done by these forces is given by

$$\left[\left(pu - \frac{\partial(pu)}{\partial x}\frac{1}{2}\delta x\right) - \left(\tau_{xx}u - \frac{\partial(\tau_{xx}u)}{\partial x}\frac{1}{2}\delta x\right)\right.$$
$$\left. - \left(pu + \frac{\partial(pu)}{\partial x}\frac{1}{2}\delta x\right) + \left(\tau_{xx}u + \frac{\partial(\tau_{xx}u)}{\partial x}\frac{1}{2}\delta x\right)\right]\delta y\delta z$$
$$+ \left[-\left(\tau_{yx}u - \frac{\partial(\tau_{yx}u)}{\partial y}\frac{1}{2}\delta y\right) + \left(\tau_{yx}u + \frac{\partial(\tau_{yx}u)}{\partial y}\frac{1}{2}\delta y\right)\right]\delta x\delta z$$
$$+ \left[-\left(\tau_{zx}u - \frac{\partial(\tau_{zx}u)}{\partial z}\frac{1}{2}\delta z\right) + \left(\tau_{zx}u + \frac{\partial(\tau_{zx}u)}{\partial z}\frac{1}{2}\delta z\right)\right]\delta x\delta y$$

The net rate of work done by these surface forces acting in the x-direction is given by

$$\left[\frac{\partial(u(-p + \tau_{xx}))}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z}\right]\delta x\delta y\delta z \tag{2.16a}$$

Surface stress components in the y- and z-direction also do work on the fluid particle. A repetition of the above process gives the additional rates of work done on the fluid particle due to the work done by these surface forces:

$$\left[\frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v(-p + \tau_{yy}))}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z}\right]\delta x\delta y\delta z \tag{2.16b}$$

and

$$\left[\frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w(-p + \tau_{zz}))}{\partial z}\right]\delta x\delta y\delta z \tag{2.16c}$$

The total rate of work done per unit volume on the fluid particle by all the surface forces is given by the sum of (2.16a–c) divided by the volume $\delta x\delta y\delta z$. The terms containing pressure can be collected together and written more compactly in vector form

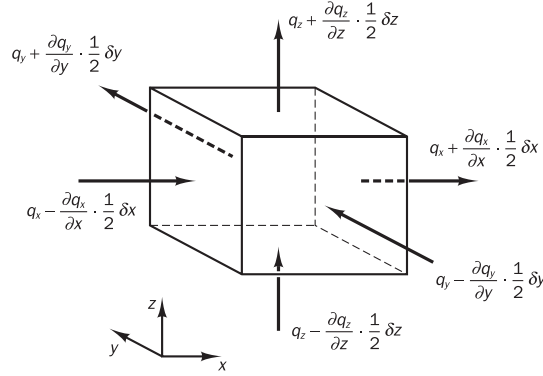
$$-\frac{\partial(up)}{\partial x} - \frac{\partial(vp)}{\partial y} - \frac{\partial(wp)}{\partial z} = -\text{div}(p\mathbf{u})$$

This yields the following **total rate of work done on the fluid particle by surface stresses**:

$$[-\text{div}(p\mathbf{u})] + \left[\frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y}\right.$$
$$\left. + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z}\right] \tag{2.17}$$

Energy flux due to heat conduction

The heat flux vector **q** has three components: q_x, q_y and q_z (Figure 2.5).

Figure 2.5 Components of the heat flux vector

The **net rate of heat transfer to the fluid particle** due to heat flow in the x-direction is given by the difference between the rate of heat input across face W and the rate of heat loss across face E:

$$\left[\left(q_x - \frac{\partial q_x}{\partial x}\frac{1}{2}\delta x\right) - \left(q_x + \frac{\partial q_x}{\partial x}\frac{1}{2}\delta x\right)\right]\delta y \delta z = -\frac{\partial q_x}{\partial x}\delta x \delta y \delta z \quad (2.18a)$$

Similarly, the net rates of heat transfer to the fluid due to heat flows in the y- and z-direction are

$$-\frac{\partial q_y}{\partial y}\delta x \delta y \delta z \quad \text{and} \quad -\frac{\partial q_z}{\partial z}\delta x \delta y \delta z \quad (2.18b\text{–}c)$$

The total rate of heat added to the fluid particle per unit volume due to heat flow across its boundaries is the sum of (2.18a–c) divided by the volume $\delta x \delta y \delta z$:

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} - \frac{\partial q_z}{\partial z} = -\text{div}\,\mathbf{q} \quad (2.19)$$

Fourier's law of heat conduction relates the heat flux to the local temperature gradient. So

$$q_x = -k\frac{\partial T}{\partial x} \quad q_y = -k\frac{\partial T}{\partial y} \quad q_z = -k\frac{\partial T}{\partial z}$$

This can be written in vector form as follows:

$$\mathbf{q} = -k \text{ grad } T \quad (2.20)$$

Combining (2.19) and (2.20) yields the final form of the **rate of heat addition to the fluid particle due to heat conduction** across element boundaries:

$$-\text{div}\,\mathbf{q} = \text{div}(k \text{ grad } T) \quad (2.21)$$

Energy equation

Thus far we have not defined the specific energy E of a fluid. Often the energy of a fluid is defined as the sum of internal (thermal) energy i, kinetic energy $\frac{1}{2}(u^2 + v^2 + w^2)$ and gravitational potential energy. This definition

takes the view that the fluid element is storing gravitational potential energy. It is also possible to regard the gravitational force as a body force, which does work on the fluid element as it moves through the gravity field.

Here we will take the latter view and include the effects of potential energy changes as a source term. As before, we define a source of energy S_E per unit volume per unit time. Conservation of energy of the fluid particle is ensured by equating the rate of change of energy of the fluid particle (2.15) to the sum of the net rate of work done on the fluid particle (2.17), the net rate of heat addition to the fluid (2.21) and the rate of increase of energy due to sources. The **energy equation** is

$$\rho\frac{DE}{Dt} = -\mathrm{div}(p\mathbf{u}) + \left[\frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z}\right] + \mathrm{div}(k\ \mathrm{grad}\ T) + S_E \tag{2.22}$$

In equation (2.22) we have $E = i + \frac{1}{2}(u^2 + v^2 + w^2)$.

Although (2.22) is a perfectly adequate energy equation it is common practice to extract the changes of the (mechanical) kinetic energy to obtain an equation for internal energy i or temperature T. The part of the energy equation attributable to the kinetic energy can be found by multiplying the x-momentum equation (2.14a) by velocity component u, the y-momentum equation (2.14b) by v and the z-momentum equation (2.14c) by w and adding the results together. It can be shown that this yields the following conservation equation for the kinetic energy:

$$\rho\frac{D[\frac{1}{2}(u^2 + v^2 + w^2)]}{Dt} = -\mathbf{u}\ .\ \mathrm{grad}\ p + u\left(\frac{\partial\tau_{xx}}{\partial x} + \frac{\partial\tau_{yx}}{\partial y} + \frac{\partial\tau_{zx}}{\partial z}\right) + v\left(\frac{\partial\tau_{xy}}{\partial x} + \frac{\partial\tau_{yy}}{\partial y} + \frac{\partial\tau_{zy}}{\partial z}\right) + w\left(\frac{\partial\tau_{xz}}{\partial x} + \frac{\partial\tau_{yz}}{\partial y} + \frac{\partial\tau_{zz}}{\partial z}\right) + \mathbf{u}\ .\ \mathbf{S}_M \tag{2.23}$$

Subtracting (2.23) from (2.22) and defining a new source term as $S_i = S_E - \mathbf{u}\ .\ \mathbf{S}_M$ yields the internal energy equation

$$\rho\frac{Di}{Dt} = -p\ \mathrm{div}\ \mathbf{u} + \mathrm{div}(k\ \mathrm{grad}\ T) + \tau_{xx}\frac{\partial u}{\partial x} + \tau_{yx}\frac{\partial u}{\partial y} + \tau_{zx}\frac{\partial u}{\partial z} + \tau_{xy}\frac{\partial v}{\partial x} + \tau_{yy}\frac{\partial v}{\partial y} + \tau_{zy}\frac{\partial v}{\partial z} + \tau_{xz}\frac{\partial w}{\partial x} + \tau_{yz}\frac{\partial w}{\partial y} + \tau_{zz}\frac{\partial w}{\partial z} + S_i \tag{2.24}$$

For the special case of an incompressible fluid we have $i = cT$, where c is the specific heat and div $\mathbf{u} = 0$. This allows us to recast (2.24) into a temperature equation

$$\rho c \frac{DT}{Dt} = \text{div}(k \text{ grad } T) + \tau_{xx}\frac{\partial u}{\partial x} + \tau_{yx}\frac{\partial u}{\partial y} + \tau_{zx}\frac{\partial u}{\partial z} + \tau_{xy}\frac{\partial v}{\partial x} + \tau_{yy}\frac{\partial v}{\partial y} + \tau_{zy}\frac{\partial v}{\partial z} + \tau_{xz}\frac{\partial w}{\partial x} + \tau_{yz}\frac{\partial w}{\partial y} + \tau_{zz}\frac{\partial w}{\partial z} + S_i \tag{2.25}$$

For compressible flows equation (2.22) is often rearranged to give an equation for the **enthalpy**. The specific enthalpy h and the specific total enthalpy h_0 of a fluid are defined as

$$h = i + p/\rho \quad \text{and} \quad h_0 = h + \tfrac{1}{2}(u^2 + v^2 + w^2)$$

Combining these two definitions with the one for specific energy E we get

$$h_0 = i + p/\rho + \tfrac{1}{2}(u^2 + v^2 + w^2) = E + p/\rho \tag{2.26}$$

Substitution of (2.26) into (2.22) and some rearrangement yields the **(total) enthalpy equation**

$$\frac{\partial(\rho h_0)}{\partial t} + \text{div}(\rho h_0 \mathbf{u}) = \text{div}(k \text{ grad } T) + \frac{\partial p}{\partial t} + \left[\frac{\partial(u\tau_{xx})}{\partial x} + \frac{\partial(u\tau_{yx})}{\partial y} + \frac{\partial(u\tau_{zx})}{\partial z} + \frac{\partial(v\tau_{xy})}{\partial x} + \frac{\partial(v\tau_{yy})}{\partial y} + \frac{\partial(v\tau_{zy})}{\partial z} + \frac{\partial(w\tau_{xz})}{\partial x} + \frac{\partial(w\tau_{yz})}{\partial y} + \frac{\partial(w\tau_{zz})}{\partial z}\right] + S_h \tag{2.27}$$

It should be stressed that equations (2.24), (2.25) and (2.27) are *not* new (extra) conservation laws but merely alternative forms of the energy equation (2.22).

2.2 Equations of state

The motion of a fluid in three dimensions is described by a system of five partial differential equations: mass conservation (2.4), x-, y- and z-momentum equations (2.14a–c) and energy equation (2.22). Among the unknowns are four thermodynamic variables: ρ, p, i and T. In this brief discussion we point out the linkage between these four variables. Relationships between the thermodynamic variables can be obtained through the assumption of **thermodynamic equilibrium**. The fluid velocities may be large, but they are usually small enough that, even though properties of a fluid particle change rapidly from place to place, the fluid can thermodynamically adjust itself to new conditions so quickly that the changes are effectively instantaneous. Thus the fluid always remains in thermodynamic equilibrium. The only exceptions are certain flows with strong shockwaves, but even some of those are often well enough approximated by equilibrium assumptions.

We can describe the state of a substance in thermodynamic equilibrium by means of just two state variables. **Equations of state** relate the other variables to the two state variables. If we use ρ and T as state variables we have state equations for pressure p and specific internal energy i:

$$p = p(\rho, T) \quad \text{and} \quad i = i(\rho, T) \tag{2.28}$$

For a **perfect gas** the following, well-known, equations of state are useful:

$$p = \rho RT \quad \text{and} \quad i = C_v T \tag{2.29}$$

The assumption of thermodynamic equilibrium eliminates all but the two thermodynamic state variables. In the flow of **compressible fluids** the equations of state provide the linkage between the energy equation on the one hand and mass conservation and momentum equations on the other. This linkage arises through the possibility of density variations as a result of pressure and temperature variations in the flow field.

Liquids and gases flowing at low speeds behave as **incompressible fluids**. Without density variations there is no linkage between the energy equation and the mass conservation and momentum equations. The flow field can often be solved by considering mass conservation and momentum equations only. The energy equation only needs to be solved alongside the others if the problem involves heat transfer.

2.3 Navier–Stokes equations for a Newtonian fluid

The governing equations contain as further unknowns the viscous stress components τ_{ij}. The most useful forms of the conservation equations for fluid flows are obtained by introducing a suitable model for the viscous stresses τ_{ij}. In many fluid flows the viscous stresses can be expressed as functions of the local deformation rate or strain rate. In three-dimensional flows the local rate of deformation is composed of the linear deformation rate and the volumetric deformation rate.

All gases and many liquids are isotropic. Liquids that contain significant quantities of polymer molecules may exhibit anisotropic or directional viscous stress properties as a result of the alignment of the chain-like polymer molecules with the flow. Such fluids are beyond the scope of this introductory course and we shall continue the development by assuming that the fluids are isotropic.

The rate of linear deformation of a fluid element has nine components in three dimensions, six of which are independent in isotropic fluids (Schlichting, 1979). They are denoted by the symbol s_{ij}. The suffix system is identical to that for stress components (see section 2.1.3). There are three linear elongating deformation components:

$$s_{xx} = \frac{\partial u}{\partial x} \quad s_{yy} = \frac{\partial v}{\partial y} \quad s_{zz} = \frac{\partial w}{\partial z} \tag{2.30a}$$

There are also six shearing linear deformation components:

$$s_{xy} = s_{yx} = \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \quad \text{and} \quad s_{xz} = s_{zx} = \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)$$

$$s_{yz} = s_{zy} = \frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \tag{2.30b}$$

The volumetric deformation is given by

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=\operatorname{div}\mathbf{u} \tag{2.30c}$$

In a **Newtonian fluid the viscous stresses are proportional to the rates of deformation**. The three-dimensional form of Newton's law of viscosity for compressible flows involves two constants of proportionality: the first (dynamic) viscosity, μ, to relate stresses to linear deformations, and the second viscosity, λ, to relate stresses to the volumetric deformation. The nine viscous stress components, of which six are independent, are

$$\tau_{xx}=2\mu\frac{\partial u}{\partial x}+\lambda\operatorname{div}\mathbf{u}\quad \tau_{yy}=2\mu\frac{\partial v}{\partial y}+\lambda\operatorname{div}\mathbf{u}\quad \tau_{zz}=2\mu\frac{\partial w}{\partial z}+\lambda\operatorname{div}\mathbf{u}$$

$$\tau_{xy}=\tau_{yx}=\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\quad \tau_{xz}=\tau_{zx}=\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)$$

$$\tau_{yz}=\tau_{zy}=\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right) \tag{2.31}$$

Not much is known about the second viscosity λ, because its effect is small in practice. For gases a good working approximation can be obtained by taking the value $\lambda=-\frac{2}{3}\mu$ (Schlichting, 1979). Liquids are incompressible so the mass conservation equation is div $\mathbf{u}=0$ and the viscous stresses are just twice the local rate of linear deformation times the dynamic viscosity.

Substitution of the above shear stresses (2.31) into (2.14a–c) yields the so-called Navier–Stokes equations, named after the two nineteenth-century scientists who derived them independently:

$$\rho\frac{Du}{Dt}=-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x}+\lambda\operatorname{div}\mathbf{u}\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]+S_{Mx} \tag{2.32a}$$

$$\rho\frac{Dv}{Dt}=-\frac{\partial p}{\partial y}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[2\mu\frac{\partial v}{\partial y}+\lambda\operatorname{div}\mathbf{u}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+S_{My} \tag{2.32b}$$

$$\rho\frac{Dw}{Dt}=-\frac{\partial p}{\partial z}+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+\frac{\partial}{\partial z}\left[2\mu\frac{\partial w}{\partial z}+\lambda\operatorname{div}\mathbf{u}\right]+S_{Mz} \tag{2.32c}$$

Often it is useful to rearrange the viscous stress terms as follows:

$$\frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x}+\lambda \operatorname{div}\mathbf{u}\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]$$

$$=\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial u}{\partial z}\right)$$

$$+\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial w}{\partial x}\right)+\frac{\partial}{\partial x}(\lambda \operatorname{div}\mathbf{u})\right]$$

$$=\operatorname{div}(\mu \operatorname{grad} u)+[s_{Mx}]$$

The viscous stresses in the y- and z-component equations can be recast in a similar manner. We clearly intend to simplify the momentum equations by 'hiding' the bracketed smaller contributions to the viscous stress terms in the momentum source. Defining a new source by

$$S_M = S_M + [s_M] \tag{2.33}$$

the **Navier–Stokes equations** can be written in the most useful form for the development of the finite volume method:

$$\boxed{\rho\frac{Du}{Dt}=-\frac{\partial p}{\partial x}+\operatorname{div}(\mu \operatorname{grad} u)+S_{Mx}} \tag{2.34a}$$

$$\boxed{\rho\frac{Dv}{Dt}=-\frac{\partial p}{\partial y}+\operatorname{div}(\mu \operatorname{grad} v)+S_{My}} \tag{2.34b}$$

$$\boxed{\rho\frac{Dw}{Dt}=-\frac{\partial p}{\partial z}+\operatorname{div}(\mu \operatorname{grad} w)+S_{Mz}} \tag{2.34c}$$

If we use the Newtonian model for viscous stresses in the internal energy equation (2.24) we obtain after some rearrangement

$$\boxed{\rho\frac{Di}{Dt}=-p \operatorname{div}\mathbf{u}+\operatorname{div}(k \operatorname{grad} T)+\Phi+S_i} \tag{2.35}$$

All the effects due to viscous stresses in this internal energy equation are described by the dissipation function Φ, which, after considerable algebra, can be shown to be equal to

$$\Phi=\mu\left\{2\left[\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial v}{\partial y}\right)^2+\left(\frac{\partial w}{\partial z}\right)^2\right]\right.$$

$$\left.+\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)^2+\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)^2+\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)^2\right\}$$

$$+\lambda(\operatorname{div}\mathbf{u})^2 \tag{2.36}$$

The dissipation function is non-negative since it only contains squared terms and represents a source of internal energy due to deformation work on the fluid particle. This work is extracted from the mechanical agency which causes the motion and converted into internal energy or heat.

2.4 Conservative form of the governing equations of fluid flow

To summarise the findings thus far, we quote in Table 2.1 the conservative or divergence form of the system of equations which governs the time-dependent three-dimensional fluid flow and heat transfer of a compressible Newtonian fluid.

Table 2.1 Governing equations of the flow of a compressible Newtonian fluid

Continuity	$\frac{\partial \rho}{\partial t} + \text{div}(\rho \mathbf{u}) = 0$	(2.4)
x-momentum	$\frac{\partial(\rho u)}{\partial t} + \text{div}(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{ grad } u) + S_{Mx}$	(2.37a)
y-momentum	$\frac{\partial(\rho v)}{\partial t} + \text{div}(\rho v \mathbf{u}) = -\frac{\partial p}{\partial y} + \text{div}(\mu \text{ grad } v) + S_{My}$	(2.37b)
z-momentum	$\frac{\partial(\rho w)}{\partial t} + \text{div}(\rho w \mathbf{u}) = -\frac{\partial p}{\partial z} + \text{div}(\mu \text{ grad } w) + S_{Mz}$	(2.37c)
Energy	$\frac{\partial(\rho i)}{\partial t} + \text{div}(\rho i \mathbf{u}) = -p \text{ div } \mathbf{u} + \text{div}(k \text{ grad } T) + \Phi + S_i$	(2.38)
Equations of state	$p = p(\rho, T)$ and $i = i(\rho, T)$	(2.28)
	e.g. perfect gas $p = \rho RT$ and $i = C_v T$	(2.29)

Momentum source S_M and dissipation function Φ are defined by (2.33) and (2.36) respectively.

It is interesting to note that the thermodynamic equilibrium assumption of section 2.2 has supplemented the five flow equations (PDEs) with two further algebraic equations. The further introduction of the Newtonian model, which expresses the viscous stresses in terms of gradients of velocity components, has resulted in a system of seven equations with seven unknowns. With an equal number of equations and unknown functions this system is mathematically closed, i.e. it can be solved provided that suitable auxiliary conditions, namely initial and boundary conditions, are supplied.

2.5 Differential and integral forms of the general transport equations

It is clear from Table 2.1 that there are significant commonalities between the various equations. If we introduce a general variable ϕ the conservative form of all fluid flow equations, including equations for scalar quantities such as temperature and pollutant concentration etc., can usefully be written in the following form:

$$\frac{\partial(\rho\phi)}{\partial t} + \text{div}(\rho\phi\mathbf{u}) = \text{div}(\Gamma \text{ grad } \phi) + S_\phi \tag{2.39}$$

In words,

Rate of increase of ϕ of fluid element	+	Net rate of flow of ϕ out of fluid element	=	Rate of increase of ϕ due to diffusion	+	Rate of increase of ϕ due to sources

Equation (2.39) is the so-called **transport equation** for property ϕ. It clearly highlights the various transport processes: the **rate of change** term and the **convective** term on the left hand side and the **diffusive** term (Γ = diffusion coefficient) and the **source** term respectively on the right hand side. In order to bring out the common features we have, of course, had to hide the terms that are not shared between the equations in the source terms. Note that equation (2.39) can be made to work for the internal energy equation by changing i into T or vice versa by means of an equation of state.

Equation (2.39) is used as the starting point for computational procedures in the finite volume method. By setting ϕ equal to 1, u, v, w and i (or T or h_0) and selecting appropriate values for diffusion coefficient Γ and source terms, we obtain special forms of Table 2.1 for each of the five PDEs for mass, momentum and energy conservation. The key step of the finite volume method, which is to be to be developed from Chapter 4 onwards, is the integration of (2.39) over a three-dimensional control volume (CV):

$$\int_{CV} \frac{\partial(\rho\phi)}{\partial t} dV + \int_{CV} \text{div}(\rho\phi\mathbf{u}) dV = \int_{CV} \text{div}(\Gamma \, \text{grad} \, \phi) dV + \int_{CV} S_\phi dV \tag{2.40}$$

The volume integrals in the second term on the left hand side, the convective term, and in the first term on the right hand side, the diffusive term, are rewritten as integrals over the entire bounding surface of the control volume by using Gauss's divergence theorem. For a vector **a** this theorem states

$$\int_{CV} \text{div}(\mathbf{a}) dV = \int_A \mathbf{n} \, . \, \mathbf{a} dA \tag{2.41}$$

The physical interpretation of **n.a** is the component of vector **a** in the direction of the vector **n** normal to surface element dA. Thus the integral of the divergence of a vector **a** over a volume is equal to the component of **a** in the direction normal to the surface which bounds the volume summed (integrated) over the entire bounding surface A. Applying Gauss's divergence theorem, equation (2.40) can be written as follows:

$$\frac{\partial}{\partial t}\left(\int_{CV} \rho\phi dV\right) + \int_A \mathbf{n} \, . \, (\rho\phi\mathbf{u}) dA = \int_A \mathbf{n} \, . \, (\Gamma \, \text{grad} \, \phi) dA + \int_{CV} S_\phi dV \tag{2.42}$$

The order of integration and differentiation has been changed in the first term on the left hand side of (2.42) to illustrate its physical meaning. This term signifies the **rate of change of the total amount of fluid property ϕ in the control volume**. The product $\mathbf{n}.\rho\phi\mathbf{u}$ expresses the flux component of property ϕ due to fluid flow along the outward normal vector **n**, so the second term on the left hand side of (2.42), the convective term, therefore is the **net rate of decrease of fluid property ϕ of the fluid element due to convection**.

A diffusive flux is positive in the direction of a negative gradient of the fluid property ϕ, i.e. along direction $-\text{grad}\ \phi$. For instance, heat is conducted in the direction of negative temperature gradients. Thus, the product $\mathbf{n} \cdot (-\Gamma\ \text{grad}\ \phi)$ is the component of diffusion flux along the outward normal vector, so out of the fluid element. Similarly, the product $\mathbf{n} \cdot (\Gamma\ \text{grad}\ \phi)$, which is also equal to $\Gamma(-\mathbf{n} \cdot (-\text{grad}\ \phi))$, can be interpreted as a positive diffusion flux in the direction of the inward normal vector $-\mathbf{n}$, i.e. into the fluid element. The first term on the right hand side of (2.42), the diffusive term, is thus associated with a flux into the element and represents the **net rate of increase of fluid property ϕ of the fluid element due to diffusion**. The final term on the right hand side of this equation gives the **rate of increase of property ϕ as a result of sources** inside the fluid element.

In words, relationship (2.42) can be expressed as follows:

Rate of increase of ϕ inside the control volume	+	Net rate of decrease of ϕ due to convection across the control volume boundaries	=	Net rate of increase of ϕ due to diffusion across the control volume boundaries	+	Net rate of creation of ϕ inside the control volume

This discussion clarifies that integration of the PDE generates a statement of the conservation of a fluid property for a finite size (macroscopic) control volume.

In steady state problems the rate of change term of (2.42) is equal to zero. This leads to the integrated form of the steady transport equation:

$$\int_A \mathbf{n} \cdot (\rho\phi\mathbf{u})\mathrm{d}A = \int_A \mathbf{n} \cdot (\Gamma\ \text{grad}\ \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \tag{2.43}$$

In time-dependent problems it is also necessary to integrate with respect to time t over a small interval Δt from, say, t until $t + \Delta t$. This yields the most general integrated form of the transport equation:

$$\int_{\Delta t} \frac{\partial}{\partial t}\left(\int_{CV} \rho\phi \mathrm{d}V\right)\mathrm{d}t + \int_{\Delta t}\int_A \mathbf{n} \cdot (\rho\phi\mathbf{u})\mathrm{d}A\mathrm{d}t = \int_{\Delta t}\int_A \mathbf{n} \cdot (\Gamma\ \text{grad}\ \phi)\mathrm{d}A\mathrm{d}t + \int_{\Delta t}\int_{CV} S_\phi \mathrm{d}V\mathrm{d}t \tag{2.44}$$

2.6 Classification of physical behaviours

Now that we have derived the conservation equations of fluid flows the time has come to turn our attention to the issue of the initial and boundary conditions that are needed in conjunction with the equations to construct a well-posed mathematical model of a fluid flow. First we distinguish two principal categories of physical behaviour:

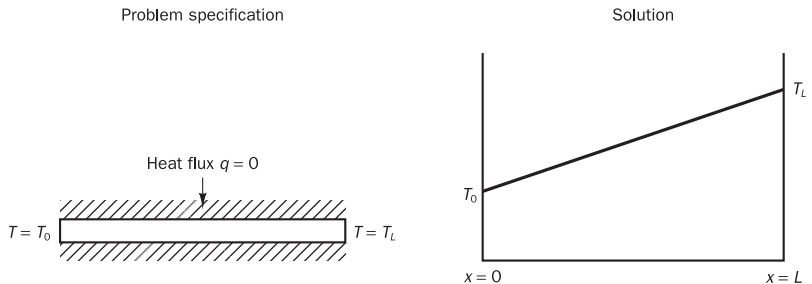
- Equilibrium problems
- Marching problems

Equilibrium problems

The problems in the first category are steady state situations, e.g. the steady state distribution of temperature in a rod of solid material or the equilibrium stress distribution of a solid object under a given applied load, as well as many steady fluid flows. These and many other steady state problems are governed by **elliptic equations**. The prototype elliptic equation is Laplace's equation, which describes irrotational flow of an incompressible fluid and steady state conductive heat transfer. In two dimensions we have

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \tag{2.45}$$

A very simple example of an equilibrium problem is the steady state heat conduction (where $\phi = T$ in equation (2.45)) in an insulated rod of metal whose ends at $x = 0$ and $x = L$ are kept at constant, but different, temperatures T_0 and T_L (Figure 2.6).

Figure 2.6 Steady state temperature distribution of an insulated rod

This problem is one-dimensional and governed by the equation $k\mathrm{d}^2T/\mathrm{d}x^2 = 0$. Under the given boundary conditions the temperature distribution in the x-direction will, of course, be a straight line. A unique solution to this and all elliptic problems can be obtained by specifying conditions on the dependent variable (here the temperature or its normal derivative the heat flux) on all the boundaries of the solution domain. Problems requiring data over the entire boundary are called **boundary-value problems**.

An important feature of elliptic problems is that a disturbance in the interior of the solution, e.g. a change in temperature due to the sudden appearance of a small local heat source, changes the solution everywhere else. Disturbance signals travel in all directions through the interior solution. Consequently, the solutions to physical problems described by elliptic equations are always smooth even if the boundary conditions are discontinuous, which is a considerable advantage to the designer of numerical methods. To ensure that information propagates in all directions, the numerical techniques for elliptic problems must allow events at each point to be influenced by all its neighbours.

Marching problems

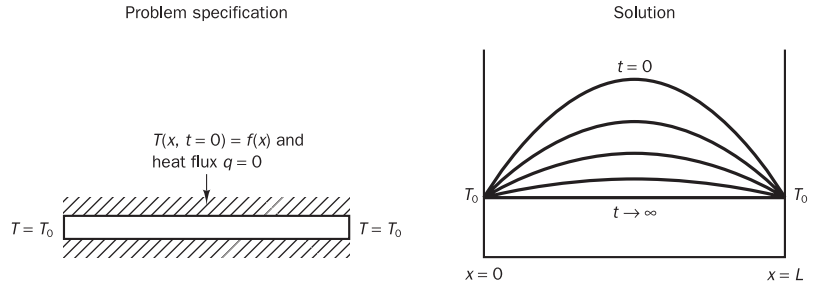
Transient heat transfer, all unsteady flows and wave phenomena are examples of problems in the second category, the marching or propagation problems. These problems are governed by **parabolic or hyperbolic equations**. However, not all marching problems are unsteady. We will see further on

that certain steady flows are described by parabolic or hyperbolic equations. In these cases the flow direction acts as a time-like co-ordinate along which marching is possible.

Parabolic equations describe time-dependent problems, which involve significant amounts of diffusion. Examples are unsteady viscous flows and unsteady heat conduction. The prototype parabolic equation is the diffusion equation

$$\frac{\partial \phi}{\partial t} = \alpha \frac{\partial^2 \phi}{\partial x^2} \tag{2.46}$$

The transient distribution of temperature (again $\phi = T$) in an insulated rod of metal whose ends at $x = 0$ and $x = L$ are kept at constant and equal temperature T_0 is governed by the diffusion equation. This problem arises when the rod cools down after an initially uniform source is switched off at time $t = 0$. The temperature distribution at the start is a parabola with a maximum at $x = L/2$ (Figure 2.7).

Figure 2.7 Transient distribution of temperature in an insulated rod

The steady state consists of a uniform distribution of temperature $T = T_0$ throughout the rod. The solution of the diffusion equation (2.46) yields the exponential decay of the initial quadratic temperature distribution. Initial conditions are needed in the entire rod and conditions on all its boundaries are required for all times $t > 0$. This type of problem is termed an **initial–boundary-value problem**.

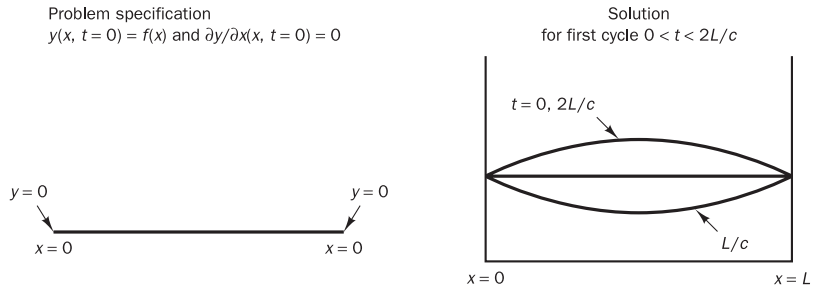
A disturbance at a point in the interior of the solution region (i.e. $0 < x < L$ and time $t_1 > 0$) can only influence events at later times $t > t_1$ (unless we allow time travel!). The solutions move forward in time and diffuse in space. The occurrence of diffusive effects ensures that the solutions are always smooth in the interior at times $t > 0$ even if the initial conditions contain discontinuities. The steady state is reached as time $t \to \infty$ and is elliptic. This change of character can be easily seen by setting $\partial\phi/\partial t = 0$ in equation (2.46). The governing equation is now equal to the one governing the steady temperature distribution in the rod.

Hyperbolic equations dominate the analysis of vibration problems. In general they appear in time-dependent processes with negligible amounts of energy dissipation. The prototype hyperbolic equation is the wave equation

$$\frac{\partial^2 \phi}{\partial t^2} = c^2 \frac{\partial^2 \phi}{\partial x^2} \tag{2.47}$$

The above form of the equation governs the transverse displacement ($\phi = y$) of a string under tension during small-amplitude vibrations and also acoustic oscillations (Figure 2.8). The constant c is the wave speed. It is relatively

straightforward to compute the fundamental mode of vibration of a string of length L using (2.47).

Figure 2.8 Vibrations of a string under tension

Solutions to wave equation (2.47) and other hyperbolic equations can be obtained by specifying two initial conditions on the displacement y of the string and one condition on all boundaries for times $t > 0$. Thus hyperbolic problems are also initial–boundary-value problems.

If the initial amplitude is given by a, the solution of this problem is

$$y(x, t) = a \cos\left(\frac{\pi ct}{L}\right) \sin\left(\frac{\pi x}{L}\right)$$

The solution shows that the vibration amplitude remains constant, which demonstrates the lack of damping in the system. This absence of damping has a further important consequence. Consider, for example, initial conditions corresponding to a near-triangular initial shape whose apex is a section of a circle with very small radius of curvature. This initial shape has a sharp discontinuity at the apex, but it can be represented by means of a Fourier series as a combination of sine waves. The governing equation is linear so each of the individual Fourier components (and also their sum) would persist in time without change of amplitude. The final result is that the discontinuity remains undiminished due to the absence of a dissipation mechanism to remove the kink in the slope.

Compressible fluid flows at speeds close to and above the speed of sound exhibit shockwaves and it turns out that the inviscid flow equations are hyperbolic at these speeds. The shockwave discontinuities are manifestations of the hyperbolic nature of such flows. Computational algorithms for hyperbolic problems are shaped by the need to allow for the possible existence of discontinuities in the interior of the solution.

It will be shown that disturbances at a point can only influence a limited region in space. The speed of disturbance propagation through an hyperbolic problem is finite and equal to the wave speed c. In contrast, parabolic and elliptic models assume infinite propagation speeds.

2.7 The role of characteristics in hyperbolic equations

Hyperbolic equations have a special behaviour, which is associated with the finite speed, namely the wave speed, at which information travels through the problem. This distinguishes hyperbolic equations from the two other types. To develop the ideas about the role of characteristic lines in hyperbolic problems we consider again a simple hyperbolic problem described by

wave equation (2.47). It can be shown (The Open University, 1984) that a change of variables to $\zeta = x - ct$ and $\eta = x + ct$ transforms the wave equation into the following standard form:

$$\frac{\partial^2 \phi}{\partial \zeta \partial \eta} = 0 \tag{2.48}$$

The transformation requires repeated application of the chain rule for differentiation to express the derivatives of equation (2.47) in terms of derivatives of the transform variables. Equation (2.48) can be solved very easily. The solution is, of course, $\phi(\zeta, \eta) = F_1(\zeta) + F_2(\eta)$, where F_1 and F_2 can be any function.

A return to the original variables yields the general solution of equation (2.47):

$$\phi(x, t) = F_1(x - ct) + F_2(x + ct) \tag{2.49}$$

The first component of the solution, function F_1, is constant if $x - ct$ is constant and hence along lines of slope $\mathrm{d}t/\mathrm{d}x = 1/c$ in the x–t plane. The second component F_2 is constant if $x + ct$ is constant, so along lines of slope $\mathrm{d}t/\mathrm{d}x = -1/c$. The lines $x - ct$ = constant and $x + ct$ = constant are called the characteristics. Functions F_1 and F_2 represent the so-called **simple wave solutions** of the problem, which are travelling waves with velocities $+c$ and $-c$ without change of shape or amplitude.

The particular forms of functions F_1 and F_2 can be obtained from the initial and boundary conditions of the problem. Let us consider a very long string ($-\infty < x < \infty$) and let the following initial conditions hold:

$$\phi(x, 0) = f(x) \quad \text{and} \quad \partial\phi/\partial t(x, 0) = g(x) \tag{2.50}$$

Combining (2.49) and (2.50) we obtain

$$F_1(x) + F_2(x) = f(x) \quad \text{and} \quad -cF_1'(x) + cF_2'(x) = g(x) \tag{2.51}$$

It can be shown (Bland, 1988) that the particular solution of wave equation (2.47) with initial conditions (2.50) is given by

$$\phi(x, t) = \frac{1}{2}[f(x - ct) + f(x + ct)] + \frac{1}{2c}\int_{x-ct}^{x+ct} g(s)\mathrm{d}s \tag{2.52}$$

Careful inspection of (2.52) shows that ϕ at point (x, t) in the solution domain depends only on the initial conditions in the interval $(x - ct, x + ct)$. It is particularly important to note that this implies that **the solution at (*x*, *t*) does not depend on initial conditions outside this interval**.

Figure 2.9 seeks to illustrate this point. The characteristics $x - ct$ = constant and $x + ct$ = constant through the point (x', t') intersect the x-axis at the points $(x' - ct', 0)$ and $(x' + ct', 0)$ respectively. The region in the x–t plane enclosed by the x-axis and the two characteristics is termed the **domain of dependence**.

In accordance with (2.52) the solution at (x', t') is influenced only by events inside the domain of dependence and not those outside. Physically this is caused by the limited propagation speed (equal to wave speed c) of mutual influences through the solution domain. Changes at the point (x', t') influence events at later times within the **zone of influence** shown in Figure 2.9, which is again bounded by the characteristics.

Figure 2.9 Domain of dependence and zone of influence for an hyperbolic problem

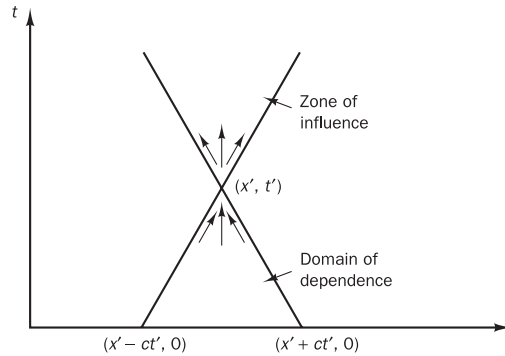
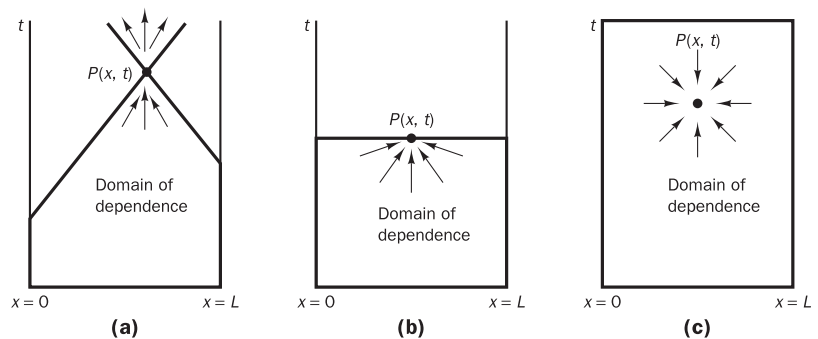



Figure 2.10a shows the situation for the vibrations of a string fixed at $x = 0$ and $x = L$. For points very close to the x-axis the domain of dependence is enclosed by two characteristics, which originate at points on the x-axis. The characteristics through points such as P intersect the problem boundaries. The domain of dependence of P is bounded by these two characteristics and the lines $t = 0$, $x = 0$ and $x = L$.

Figure 2.10 Domains of dependence for the (a) hyperbolic, (b) parabolic and (c) elliptic problem

The shape of the domains of dependence (see Figures 2.10b and c) in parabolic and elliptic problems is different because the speed of information travel is assumed to be infinite. The bold lines which demarcate the boundaries of each domain of dependence give the regions for which initial and/or boundary conditions are needed in order to be able to generate a solution at the point $P(x, t)$ in each case.

The way in which changes at one point affect events at other points depends on whether a physical problem represents a steady state or a transient phenomenon and whether the propagation speed of disturbances is finite or infinite. This has resulted in a classification of physical behaviours, and hence attendant PDEs, into elliptic, parabolic and hyperbolic problems. The distinguishing features of each of the categories were illustrated by considering three simple prototype second-order equations. In the following sections we will discuss methods of classifying more complex PDEs and briefly state the limitations of the computational methods that will be developed later in this text in terms of the classification of the flow problems to be solved. A summary of the main features that have been identified so far is given in Table 2.2.

Table 2.2 Classification of physical behaviours

Problem type	*Equation type*	*Prototype equation*	*Conditions*	*Solution domain*	*Solution smoothness*
Equilibrium problems	Elliptic	$\text{div grad } \phi = 0$	Boundary conditions	Closed domain	Always smooth
Marching problems with dissipation	Parabolic	$\dfrac{\partial \phi}{\partial t} = \alpha \text{ div grad } \phi$	Initial and boundary conditions	Open domain	Always smooth
Marching problems without dissipation	Hyperbolic	$\dfrac{\partial^2 \phi}{\partial t^2} = c^2 \text{ div grad } \phi$	Initial and boundary conditions	Open domain	May be discontinuous

2.8 Classification method for simple PDEs

A practical method of classifying PDEs is developed for a general second-order PDE in two co-ordinates x and y. Consider

$$a\frac{\partial^2 \phi}{\partial x^2} + b\frac{\partial^2 \phi}{\partial x \partial y} + c\frac{\partial^2 \phi}{\partial y^2} + d\frac{\partial \phi}{\partial x} + e\frac{\partial \phi}{\partial y} + f\phi + g = 0 \tag{2.53}$$

At first we shall assume that the equation is linear and a, b, c, d, e, f and g are constants.

The classification of a PDE is governed by the behaviour of its highest-order derivatives, so we need only consider the second-order derivatives. The class of a second-order PDE can be identified by searching for possible simple wave solutions. If they exist this indicates a hyperbolic equation. If not the equation is parabolic or elliptic.

Simple wave solutions occur if the characteristic equation (2.54) below has two real roots:

$$a\left(\frac{\mathrm{d}y}{\mathrm{d}x}\right)^2 - b\left(\frac{\mathrm{d}y}{\mathrm{d}x}\right) + c = 0 \tag{2.54}$$

The existence or otherwise of roots of the characteristic equation depends on the value of discriminant ($b^2 - 4ac$). Table 2.3 outlines the three cases.

Table 2.3 Classification of linear second-order PDEs

$b^2 - 4ac$	*Equation type*	*Characteristics*
> 0	Hyperbolic	Two real characteristics
$= 0$	Parabolic	One real characteristic
< 0	Elliptic	No characteristics

It is left as an exercise for the reader to verify the nature of the three prototype PDEs in section 2.6 by evaluating the discriminant.

The classification method by searching for the roots of the characteristic equation also applies if the coefficients a, b and c are functions of x and y or if the equation is non-linear. In the latter case a, b and c may be functions of dependent variable ϕ or its first derivatives. It is now possible that the

equation type differs in various regions of the solution domain. As an example we consider the following equation:

$$y\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \tag{2.55}$$

We look at the behaviour within the region $-1 < y < 1$. Hence $a = a(x, y) = y$, $b = 0$ and $c = 1$. The value of discriminant $(b^2 - 4ac)$ is equal to $-4y$. We need to distinguish three cases:

- If $y < 0$: $b^2 - 4ac > 0$ so the equation is hyperbolic
- If $y = 0$: $b^2 - 4ac = 0$ so the equation is parabolic
- If $y > 0$: $b^2 - 4ac < 0$ and hence the equation is elliptic

Equation (2.55) is of mixed type. The equation is locally hyperbolic, parabolic or elliptic depending on the value of y. For the non-linear case similar remarks apply. The classification of the PDE depends on the local values of a, b and c.

Second-order PDEs in N independent variables $(x_1, x_2, \ldots, x_N)$ can be classified by rewriting them first in the following form with $A_{jk} = A_{kj}$:

$$\sum_{j=1}^{N}\sum_{k=1}^{N} A_{jk}\frac{\partial^2 \phi}{\partial x_j \partial x_k} + H = 0 \tag{2.56}$$

Fletcher (1991) explains that the equation can be classified on the basis of the eigenvalues of a matrix with entries A_{jk}. Hence we need to find values for λ for which

$$\det[A_{jk} - \lambda I] = 0 \tag{2.57}$$

The classification rules are:

- if any eigenvalue $\lambda = 0$: the equation is parabolic
- if all eigenvalues $\lambda \neq 0$ and they are all of the same sign: the equation is elliptic
- if all eigenvalues $\lambda \neq 0$ and all but one are of the same sign: the equation is hyperbolic

In the cases of Laplace's equation, the diffusion equation and the wave equation it is simple to verify that this method yields the same results as the solution of characteristic equation (2.54).

2.9 Classification of fluid flow equations

Systems of first-order PDEs with more than two independent variables are similarly cast in matrix form. Their classification involves finding eigenvalues of the resulting matrix. Systems of second-order PDEs or mixtures of first- and second-order PDEs can also be classified with this method. The first stage of the method involves the introduction of auxiliary variables, which express each second-order equation as first-order equations. Care must be taken to select the auxiliary variables in such a way that the resulting matrix is non-singular.

The Navier–Stokes equation and its reduced forms can be classified using such a matrix approach. The details are beyond the scope of this introduction to the subject. We quote the main results in Table 2.4 and refer the interested reader to Fletcher (1991) for a full discussion.

Table 2.4 Classification of the main categories of fluid flow

	Steady flow	*Unsteady flow*
Viscous flow	Elliptic	Parabolic
Inviscid flow	$M < 1$, elliptic $M > 1$, hyperbolic	Hyperbolic
Thin shear layers	Parabolic	Parabolic

The classifications in Table 2.4 are the 'formal' classifications of the flow equations. In practice many fluid flows behave in a complex way. The steady Navier–Stokes equations and the energy (or enthalpy) equations are formally elliptic and the unsteady equations are parabolic.

The mathematical classification of inviscid flow equations is different from the Navier–Stokes and energy equations due to the complete absence of the (viscous) higher-order terms. The classification of the resulting equation set depends on the extent to which fluid compressibility plays a role and hence on the magnitude of the Mach number M. The elliptic nature of inviscid flows at Mach numbers below 1 originates from the action of pressure. If $M < 1$ the pressure can propagate disturbances at the speed of sound, which is greater than the flow speed. But if $M > 1$ the fluid velocity is greater than the propagation speed of disturbances and the pressure is unable to influence events in the upstream direction. Limitations on the zone of influence are a key feature of hyperbolic phenomena, so the supersonic inviscid flow equations are hyperbolic. Below, we will see a simple example that demonstrates this behaviour.

In thin shear layer flows all velocity derivatives in the flow (x- and z-) direction are much smaller than those in the cross-stream (y-) direction. Boundary layers, jets, mixing layers and wakes as well as fully developed duct flows fall within this category. In these conditions the governing equations contain only one (second-order) diffusion term and are therefore classified as parabolic.

As an illustration of the complexities which may arise in inviscid flows we analyse the potential equation which governs steady, isentropic, inviscid, compressible flow past a slender body (Shapiro, 1953) with a free stream Mach number M_∞:

$$(1 - M_\infty^2)\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \qquad (2.58)$$

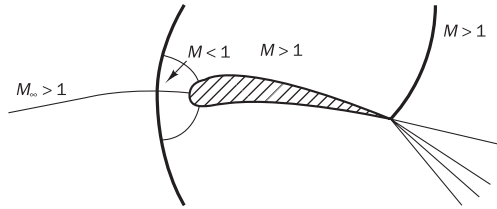
Taking $x_1 = x$ and $x_2 = y$ in equation (2.56) we have matrix elements $A_{11} = 1 - M_\infty^2$, $A_{12} = A_{21} = 0$ and $A_{22} = 1$. To classify the equation we need to solve

$$\det\begin{vmatrix} (1 - M_\infty^2) - \lambda & 0 \\ 0 & 1 - \lambda \end{vmatrix} = 0$$

The two solutions are $\lambda_1 = 1$ and $\lambda_2 = 1 - M_\infty^2$. If the free stream Mach number is smaller than 1 (subsonic flow) both eigenvalues are greater than zero and the flow is elliptic. If the Mach number is greater than 1 (supersonic flow) the second eigenvalue is negative and the flow is hyperbolic. The reader is left to demonstrate that these results are identical to those obtained by considering the discriminant of characteristic equation (2.54).

It is interesting to note that we have discovered an instance of hyperbolic behaviour in a steady flow where both independent variables are space coordinates. The flow direction behaves in a time-like manner in hyperbolic inviscid flows and also in the parabolic thin shear layers. These problems are of the marching type and flows can be computed by marching in the time-like direction of increasing x.

The above example shows the dependence of the classification of compressible flows on the parameter M_∞. The general equations of inviscid, compressible flow (the Euler equations) exhibit similar behaviour, but the classification parameter is now the local Mach number M. This complicates matters greatly when flows around and above $M = 1$ are to be computed. Such flows may contain shockwave discontinuities and regions of subsonic (elliptic) flow and supersonic (hyperbolic) flow, whose exact locations are not known a priori. Figure 2.11 is a sketch of the flow around an aerofoil at a Mach number somewhat greater than 1.

Figure 2.11 Sketch of flow around an aerofoil at supersonic free stream speed

2.10 Auxiliary conditions for viscous fluid flow equations

The complicated mixture of elliptic, parabolic and hyperbolic behaviours has implications for the way in which boundary conditions enter into a flow problem, in particular at locations where flows are bounded by fluid boundaries. Unfortunately few theoretical results regarding the range of permissible boundary conditions are available for compressible flows. CFD practice is guided here by physical arguments and the success of its simulations. The **boundary conditions for a compressible viscous flow** are given in Table 2.5.

In the table subscripts n and t indicate directions normal (outward) and tangential to the boundary respectively and F are the given surface stresses.

Table 2.5 Boundary conditions for compressible viscous flow

Initial conditions for unsteady flows:

- Everywhere in the solution region ρ, $\mathbf{u}$ and T must be given at time $t = 0$.

Boundary conditions for unsteady and steady flows:

- On solid walls: $\mathbf{u} = \mathbf{u}_w$ (no-slip condition)
  $T = T_w$ (fixed temperature) or $k\partial T/\partial n = -q_w$ (fixed heat flux)
- On fluid boundaries: inlet: ρ, $\mathbf{u}$ and T must be known as a function of position
  outlet: $-p + \mu\partial u_n/\partial n = F_n$ and $\mu\partial u_t/\partial n = F_t$ (stress continuity)

It is unnecessary to specify outlet or solid wall boundary conditions for the density because of the special character of the continuity equation, which describes the changes of density experienced by a fluid particle along its path for a known velocity field. At the inlet the density needs to be known. Everywhere else the density emerges as part of the solution and no boundary values need to be specified. For an **incompressible viscous flow** there are no conditions on the density, but all the other above conditions apply without modification.

Commonly outflow boundaries are positioned at locations where the flow is approximately unidirectional and where surface stresses take known values. For high Reynolds number flows far from solid objects in an external flow or in fully developed flow out of a duct, there is no change in any of the velocity components in the direction across the boundary and $F_n = -p$ and $F_t = 0$. This gives the outflow condition that is almost universally used in the finite volume method:

$$\boxed{\text{specified pressure, } \partial u_n/\partial n = 0 \text{ and } \partial T/\partial n = 0}$$

Gresho (1991) reviewed the intricacies of open boundary conditions in incompressible flow and stated that there are some 'theoretical concerns' regarding open boundary conditions which use $\partial u_n/\partial n = 0$; however, its success in CFD practice left him to recommend it as the simplest and cheapest form when compared with theoretically more satisfying selections.

Figure 2.12 illustrates the application of boundary conditions for a typical internal and external viscous flow.

General-purpose CFD codes also often include inlet and outlet pressure boundary conditions. The pressures are set at fixed values and sources and sinks of mass are placed on the boundaries to carry the correct mass flow into and out of the solution zone across the constant pressure boundaries. Furthermore, symmetric and cyclic boundary conditions are supplied to take advantage of special geometrical features of the solution region:

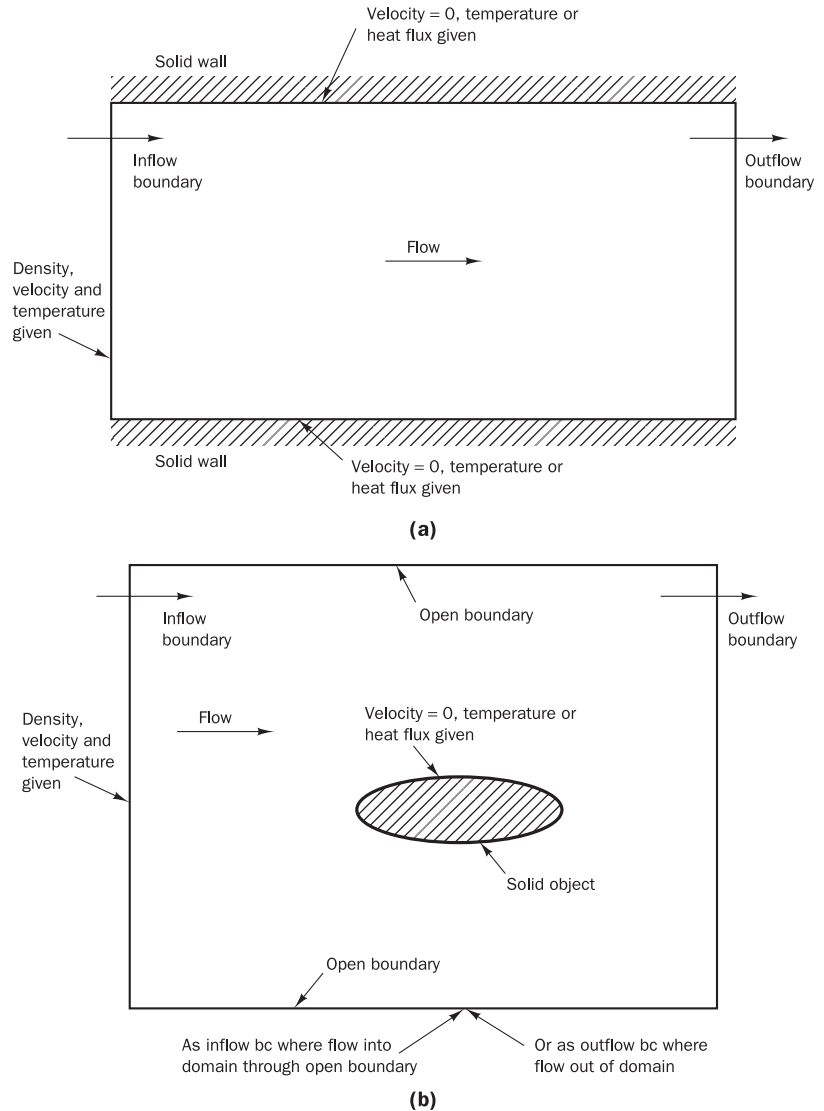
- Symmetry boundary condition: $\partial\phi/\partial n = 0$
- Cyclic boundary condition: $\phi_1 = \phi_2$

Figure 2.13 shows typical boundary geometries for which symmetry and cyclic boundary conditions (bc) may be useful.

2.11 Problems in transonic and supersonic compressible flows

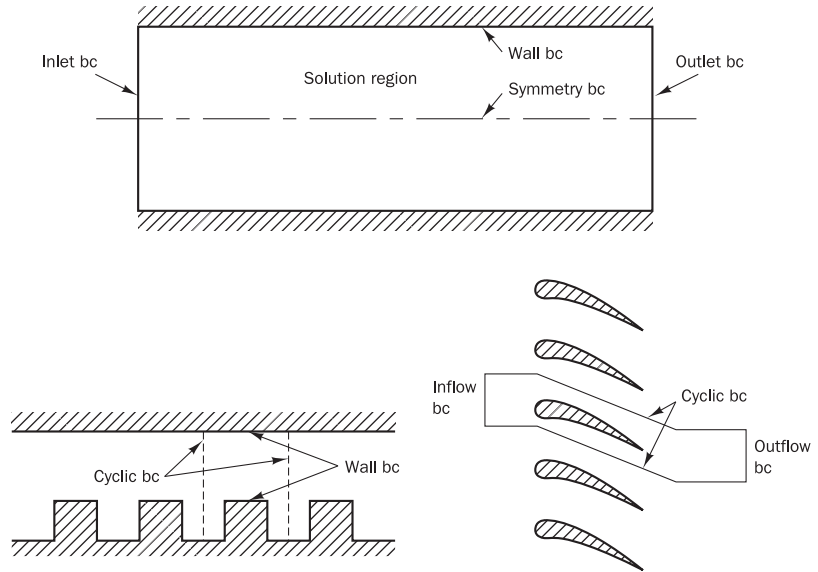
Difficulties arise when calculating flows at speeds near to and above the speed of sound. At these speeds the Reynolds number is usually very high and the viscous regions in the flow are usually very thin. The flow in a large part of the solution region behaves as an effectively inviscid fluid. This gives rise to problems in external flows, because the part of the flow where the outer boundary conditions are applied behaves in an inviscid way, which differs from the (viscous) region of flow on which the overall classification is based.

The standard SIMPLE pressure correction algorithm for finite volume calculations (see section 6.4) needs to be modified. The transient version of the algorithm needs to be adopted to make use of the favourable character of parabolic/hyperbolic procedures. To cope with the appearance of shock-waves in the solution interior and with reflections from the domain boundaries, artificial damping needs to be introduced. It is further necessary to

Figure 2.12 (a) Boundary conditions for an internal flow problem; (b) boundary condition for external flow problem

ensure that the limited domain of dependence of effectively inviscid (hyperbolic) flows at Mach numbers greater than 1 is adequately modelled. Issa and Lockwood (1977) and McGuirk and Page (1990) gave lucid papers that identify the main issues relevant to the finite volume method.

Open (far field) boundary conditions give the most serious problems for the designer of general-purpose CFD codes. Subsonic inviscid compressible flow equations require fewer inlet conditions (normally only ρ and $\mathbf{u}$ are specified) than viscous flow equations and only one outlet condition (typically specified pressure). Supersonic inviscid compressible flows require the same number of inlet boundary conditions as viscous flows, but do not admit any outflow boundary conditions because the flow is hyperbolic.

Without knowing a great deal about the flow before solving a problem it is very difficult to specify the precise number and nature of the allowable

Figure 2.13 Examples of flow boundaries with symmetry and cyclic conditions

boundary conditions on any fluid/fluid boundary in the far field. Issa and Lockwood's work (1977) reported the solution of a shock/boundary layer interaction problem where part of the far field boundary conditions are obtained from an inviscid solution performed prior to the viscous solution. The usual (viscous) outlet condition $\partial(\rho u_n)/\partial n = 0$ is applied on the remainder of the far field boundary.

Fletcher (1991) noted that under-specification of boundary conditions normally leads to failure to obtain a unique solution. Over-specification, however, gives rise to flow solutions with severe and unphysical 'boundary layers' close to the boundary where the condition is applied.

If the location of the outlet or far field boundaries is chosen far enough away from the region of interest within the solution domain it is possible to get physically meaningful results. Most careful solutions test the sensitivity of the interior solution to the positioning of outflow and far field boundaries. If results do not change in the interior, the boundary conditions are 'transparent' and the results are acceptable.

These complexities make it very difficult for general-purpose finite volume CFD codes to cope with general subsonic, transonic and/or supersonic viscous flows. Although all commercially available codes claim to be able to make computations in all flow regimes, they perform most effectively at Mach numbers well below 1 as a consequence of all the problems outlined above.

2.12 Summary

We have derived the complete set of governing equations of fluid flow from basic conservation principles. The thermodynamic equilibrium assumption and the Newtonian model of viscous stresses were enlisted to close the system mathematically. Since no particular assumptions were made with regard to the viscosity, it is straightforward to accommodate a variable viscosity that is dependent on local conditions. This facilitates the inclusion of fluids with

temperature-dependent viscosity and those with non-Newtonian characteristics within the framework of equations.

We have identified a common differential form for all the flow equations, the so-called transport equation, and developed integrated forms which are central to the finite volume CFD method:

For steady state processes

$$\boxed{\int_A \mathbf{n} \, . \, (\rho\phi\mathbf{u}) \mathrm{d}A = \int_A \mathbf{n} \, . \, (\Gamma \; \mathrm{grad} \; \phi) \mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V} \tag{2.43}$$

and for time-dependent processes

$$\boxed{\begin{aligned} &\int_{\Delta t} \frac{\partial}{\partial t}\left(\int_{CV} \rho\phi \mathrm{d}V\right) \mathrm{d}t + \int_{\Delta t}\int_A \mathbf{n} \, . \, (\rho\phi\mathbf{u}) \mathrm{d}A \; \mathrm{d}t \\ &\qquad = \int_{\Delta t}\int_A \mathbf{n} \, . \, (\Gamma \; \mathrm{grad} \; \phi) \mathrm{d}A \; \mathrm{d}t + \int_{\Delta t}\int_{CV} S_\phi \mathrm{d}V \; \mathrm{d}t \end{aligned}} \tag{2.44}$$

The auxiliary conditions – initial and boundary conditions – needed to solve a fluid flow problem were also discussed. It emerged that there are three types of distinct physical behaviour – elliptic, parabolic and hyperbolic – and the governing fluid flow equations were formally classified. Problems with this formal classification were identified as resulting from: (i) boundary-layer-type behaviour in flows at high Reynolds numbers and (ii) compressibility effects at Mach numbers around and above 1. These lead to severe difficulties in the specification of boundary conditions for completely general-purpose CFD procedures working at any Reynolds number and Mach number.

Experience with the finite volume method has yielded a set of auxiliary conditions that give physically realistic flow solutions in many industrially relevant problems. The most complete problem specification includes, in addition to initial values of all flow variables, the following boundary conditions:

- Complete specification of the distribution of all variables ϕ (except pressure) at all **inlets** to the flow domain of interest
- Specification of pressure at one location inside the flow domain
- Set gradient of all variables ϕ to zero in the flow direction at suitably positioned **outlets**
- Specification of all variables ϕ (except pressure and density) or their normal gradients at solid walls

Chapter three Turbulence and its modelling

All flows encountered in engineering practice, simple ones, such as two-dimensional jets, wakes, pipe flows and flat plate boundary layers, and more complicated three-dimensional ones, become unstable above a certain Reynolds number (UL/ν where U and L are characteristic velocity and length scales of the mean flow and ν is the kinematic viscosity). At low Reynolds numbers flows are laminar. At higher Reynolds numbers flows are observed to become turbulent. A chaotic and random state of motion develops in which the velocity and pressure change continuously with time within substantial regions of flow.

Flows in the laminar regime are completely described by the equations developed in Chapter 2. In simple cases the continuity and Navier–Stokes equations can be solved analytically (Schlichting, 1979). More complex flows can be tackled numerically with CFD techniques such as the finite volume method without additional approximations.

Many, if not most, flows of engineering significance are turbulent, so the turbulent flow regime is not just of theoretical interest. Fluid engineers need access to viable tools capable of representing the effects of turbulence. This chapter gives a brief introduction to the physics of turbulence and to its modelling in CFD.

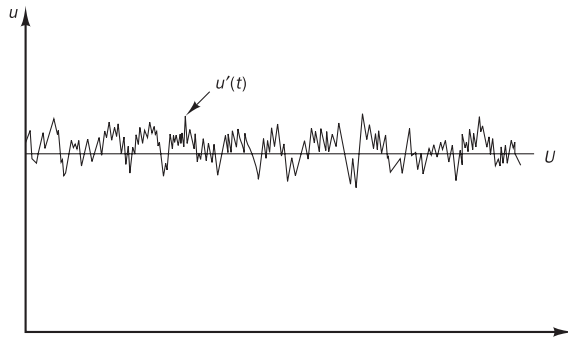
In sections 3.1 and 3.2, the nature of turbulent flows and the physics of the transition from laminar flow to turbulence are examined. In section 3.3 we give formal definitions for the most common descriptors of a turbulent flow, and in section 3.4 the characteristics of some simple two-dimensional turbulent flows are described. Next, in section 3.5, the consequences of the appearance of the fluctuations associated with turbulence on the time-averaged Navier–Stokes equations are analysed. The velocity fluctuations are found to give rise to additional stresses on the fluid, the so-called Reynolds stresses. The main categories of models for these extra stress terms are given in section 3.6. The most widely used category of classical turbulence models is discussed in section 3.7. In section 3.8 we review large eddy simulations (LES) and in section 3.9 we give a brief summary of direct numerical simulation (DNS).

3.1 What is turbulence?

First we take a brief look at the main characteristics of turbulent flows. The Reynolds number of a flow gives a measure of the relative importance of inertia forces (associated with convective effects) and viscous forces. In experiments on fluid systems it is observed that at values below the so-called critical Reynolds number Re_{crit} the flow is smooth and adjacent layers

of fluid slide past each other in an orderly fashion. If the applied boundary conditions do not change with time the flow is steady. This regime is called **laminar** flow.

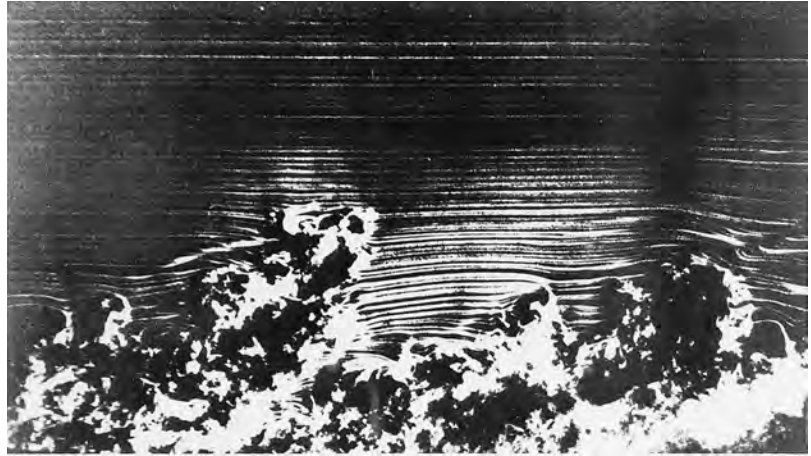
At values of the Reynolds number above Re_{crit} a complicated series of events takes place which eventually leads to a radical change of the flow character. In the final state the flow behaviour is random and chaotic. The motion becomes intrinsically unsteady even with constant imposed boundary conditions. The velocity and all other flow properties vary in a random and chaotic way. This regime is called **turbulent** flow. A typical point velocity measurement might exhibit the form shown in Figure 3.1.

Figure 3.1 Typical point velocity measurement in turbulent flow

The **random** nature of a turbulent flow precludes an economical description of the motion of all the fluid particles. Instead the velocity in Figure 3.1 is decomposed into a steady mean value U with a fluctuating component $u'(t)$ superimposed on it: $u(t) = U + u'(t)$. This is called the **Reynolds decomposition**. A turbulent flow can now be characterised in terms of the mean values of flow properties (U, V, W, P etc.) and some statistical properties of their fluctuations (u', v', w', p' etc.). We give formal definitions of the mean and the most common statistical descriptors of the fluctuations in section 3.3.

Even in flows where the mean velocities and pressures vary in only one or two space dimensions, turbulent fluctuations always have a **three-dimensional** spatial character. Furthermore, visualisations of turbulent flows reveal rotational flow structures, so-called turbulent eddies, with a **wide range of length scales**. Figure 3.2, which depicts a cross-sectional view of a turbulent boundary layer on a flat plate, shows eddies whose length scale is comparable with that of the flow boundaries as well as eddies of intermediate and small size.

Particles of fluid which are initially separated by a long distance can be brought close together by the eddying motions in turbulent flows. As a consequence, heat, mass and momentum are very effectively exchanged. For example, a streak of dye which is introduced at a point in a turbulent flow will rapidly break up and be dispersed right across the flow. Such **effective mixing** gives rise to high values of diffusion coefficients for mass, momentum and heat.

The largest turbulent eddies interact with and extract energy from the mean flow by a process called **vortex stretching**. The presence of mean velocity gradients in sheared flows distorts the rotational turbulent eddies. Suitably aligned eddies are stretched because one end is forced to move faster than the other.

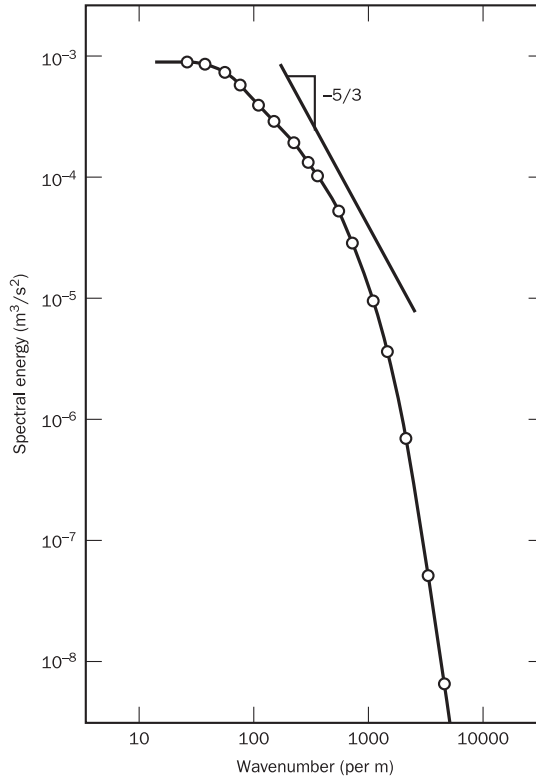
Figure 3.2 Visualisation of a turbulent boundary layer
Source: Van Dyke (1982)

The characteristic velocity ϑ and characteristic length ℓ of the larger eddies are of the same order as the velocity scale U and length scale L of the mean flow. Hence a 'large eddy' Reynolds number $Re_\ell = \vartheta\ell/\nu$ formed by combining these eddy scales with the kinematic viscosity will be large in all turbulent flows, since it is not very different in magnitude from UL/ν, which itself is large. This suggests that these large eddies are dominated by inertia effects and viscous effects are negligible.

The large eddies are therefore effectively inviscid, and angular momentum is conserved during vortex stretching. This causes the rotation rate to increase and the radius of their cross-sections to decrease. Thus the process creates motions at smaller transverse length scales and also at smaller time scales. The stretching work done by the mean flow on the large eddies during these events provides the energy which maintains the turbulence.

Smaller eddies are themselves stretched strongly by somewhat larger eddies and more weakly with the mean flow. In this way the kinetic energy is handed down from large eddies to progressively smaller and smaller eddies in what is termed the **energy cascade**. All the fluctuating properties of a turbulent flow contain energy across a wide range of frequencies or wavenumbers ($= 2\pi f/U$ where f is the frequency). This is demonstrated in Figure 3.3, which gives the energy spectrum of turbulence downstream of a grid.

The spectral energy $E(\kappa)$ is shown as a function of the wavenumber $\kappa = 2\pi/\lambda$, where λ is the wavelength of the eddies. The spectral energy $E(\kappa)$ (units m^3/s^2) is the kinetic energy per unit mass and per unit wavenumber of fluctuations around the wavenumber κ. The diagram shows that the energy content peaks at the low wavenumbers, so the larger eddies are the most energetic. They acquire their energy through strong interactions with the mean flow. The value of $E(\kappa)$ rapidly decreases as the wavenumber increases, so the smallest eddies have the lowest energy content.

The smallest scales of motion in a turbulent flow (lengths of the order of 0.1 to 0.01 mm and frequencies around 10 kHz in typical turbulent engineering flows) are dominated by viscous effects. The Reynolds number Re_η of the smallest eddies based on their characteristic velocity υ and characteristic length η is equal to 1, $Re_\eta = \upsilon\eta/\nu = 1$, so the smallest scales present in a turbulent flow are those for which the inertia and viscous effects

Figure 3.3 Energy spectrum of turbulence behind a grid

are of equal strength. These scales are named the Kolmogorov microscales after the Russian scientist who carried out groundbreaking work on the structure of turbulence in the 1940s. At these scales work is performed against the action of viscous stresses, so that the energy associated with small-scale eddy motions is dissipated and converted into thermal internal energy. This dissipation results in **increased energy losses** associated with turbulent flows.

Dimensional analysis can be used to obtain ratios of the length, time and velocity scales of the small and large eddies. The Kolmogorov microscales can be expressed in terms of the rate of energy dissipation of a turbulent flow and the fluid viscosity, which uses the notion that in every turbulent flow the rate of production of turbulent energy has to be broadly in balance with its rate of dissipation to prevent unlimited growth of turbulence energy. This yields the following order of magnitude estimates of the ratios of small length, time and velocity scales η, τ, υ and large length, time and velocity scales ℓ, T, ϑ (Tennekes and Lumley, 1972; Reynolds in Lumley, 1989):

$$\text{Length-scale ratio} \quad \frac{\eta}{\ell} \approx Re_\ell^{-3/4} \tag{3.1a}$$

$$\text{Time-scale ratio} \quad \frac{\tau}{T} \approx Re_\ell^{-1/2} \tag{3.1b}$$

$$\text{Velocity-scale ratio} \quad \frac{\upsilon}{\vartheta} \approx Re_\ell^{-1/4} \tag{3.1c}$$

Typical values of Re_ℓ might be 10^3–10^6, so the length, time and velocity scales associated with small dissipating eddies are much smaller than those of large, energetic eddies, and the difference – the so-called scale separation – increases as Re_ℓ increases.

The behaviour of the large eddies should be independent of viscosity and should depend on the velocity scale ϑ and length scale ℓ. Thus, on dimensional grounds we would expect that the spectral energy content of these eddies should behave as follows: $E(\kappa) \propto \vartheta^2\ell$, where $\kappa = 1/\ell$. Since the length scale ℓ is related to the length scale of turbulence producing processes – for example, boundary layer thickness δ, obstacle width L, surface roughness height k_s – we expect the structure of the **largest eddies** to be highly **anisotropic** (i.e. the fluctuations are different in different directions) and strongly affected by the problem boundary conditions.

Kolmogorov argued that the structure of the smallest eddies and, hence, their spectral energy $E(\kappa = 1/\eta)$ should only depend on the rate of dissipation of turbulent energy ε (units m^2/s^3) and the kinematic viscosity of the fluid ν. Dimensional analysis yields the following proportionality relationship for the spectral energy: $E(\kappa = 1/\eta) \propto \nu^{5/4}\varepsilon^{1/4}$. Thus, the spectral energy $E(\kappa)$ of the smallest eddies only depends on the problem through the *rate* of energy dissipation and is not linked to other problem variables. The diffusive action of viscosity tends to smear out directionality at small scales. At high mean flow Reynolds numbers the **smallest eddies** in a turbulent flow are, therefore, **isotropic** (non-directional).

Finally, Kolmogorov derived the universal spectral properties of eddies of intermediate size, which are sufficiently large for their behaviour to be unaffected by viscous action (as the larger eddies), but sufficiently small that the details of their behaviour can be expressed as a function of the rate of energy dissipation ε (as the smallest eddies). The appropriate length scale for these eddies is $1/\kappa$, and he found that the spectral energy of these eddies – the inertial subrange – satisfies the following relationship: $E(\kappa) = \alpha\kappa^{-5/3}\varepsilon^{2/3}$. Measurements showed that the constant $\alpha \approx 1.5$. Figure 3.3 includes a line with a slope of $-5/3$, indicating that, for the measurements shown, the scale separation is insufficient for a clear inertial subrange. Overlap between the large and small eddies is located somewhere around $\kappa \approx 1000$.

3.2 Transition from laminar to turbulent flow

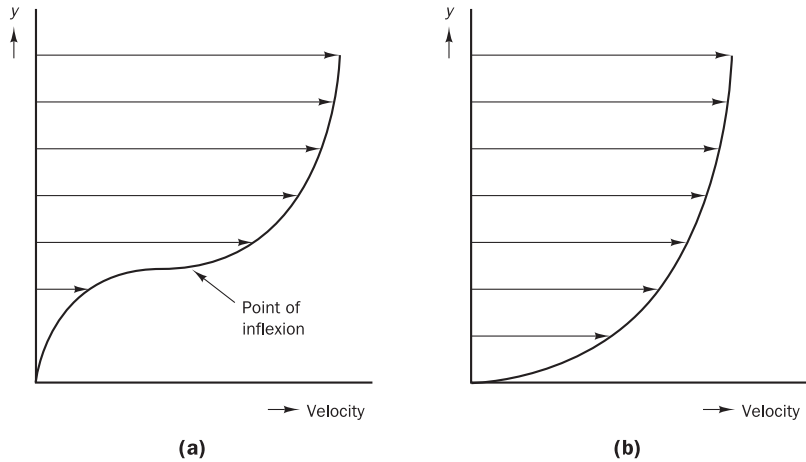
The initial cause of the **transition** to turbulence can be explained by considering the stability of laminar flows to small disturbances. A sizeable body of theoretical work is devoted to the analysis of the inception of transition: **hydrodynamic instability**. In many relevant instances the transition to turbulence is associated with sheared flows. Linear hydrodynamic stability theory seeks to identify conditions which give rise to amplification of disturbances. Of particular interest in an engineering context is the prediction of the values of the Reynolds numbers $Re_{x,crit}$ ($= Ux_{crit}/\nu$) at which disturbances are amplified and $Re_{x,tr}$ ($= Ux_{tr}/\nu$) at which transition to fully turbulent flow takes place.

A mathematical discussion of the theory is beyond the scope of this brief introduction. White (1991) gave a useful overview of theory and experiments. The subject matter is fairly complex but its confirmation has led to a series of experiments which reveal the physical processes causing transition from laminar to turbulent flow. Most of our knowledge stems from work on

two-dimensional incompressible flows. All such flows are sensitive to two-dimensional disturbances with a relatively long wavelength, several times the transverse distance over which velocity changes take place (e.g. six times the thickness of a flat plate boundary layer).

Hydrodynamic stability of laminar flows

Two fundamentally different instability mechanisms operate, which are associated with the shape of the two-dimensional laminar velocity profile of the base flow. Flows with a velocity distribution which contains a point of inflexion as shown in Figure 3.4a are always unstable with respect to infinitesimal disturbances if the Reynolds number is large enough. This instability was first identified by making an inviscid assumption in the equations describing the evolution of the disturbances. Subsequent refinement of the theory by inclusion of the effect of viscosity changed its results very little, so this type of instability is known as **inviscid instability**. Velocity profiles of the type shown in Figure 3.4a are associated with jet flows, mixing layers and wakes and also with boundary layers over flat plates under the influence of an adverse pressure gradient ($\partial p/\partial x > 0$). The role of viscosity is to dampen out fluctuations and stabilise the flow at low Reynolds numbers.

Figure 3.4 Velocity profiles susceptible to (a) inviscid instability and (b) viscous instability

Flows with laminar velocity distributions without a point of inflexion such as the profile shown in Figure 3.4b are susceptible to **viscous instability**. The approximate inviscid theory predicts unconditional stability for these velocity profiles, which are invariably associated with flows near solid walls such as pipe, channel and boundary layer flows without adverse pressure gradients ($\partial p/\partial x \leq 0$). Viscous effects play a more complex role providing damping at low and high Reynolds numbers, but contributing to the destabilisation of the flows at intermediate Reynolds numbers.

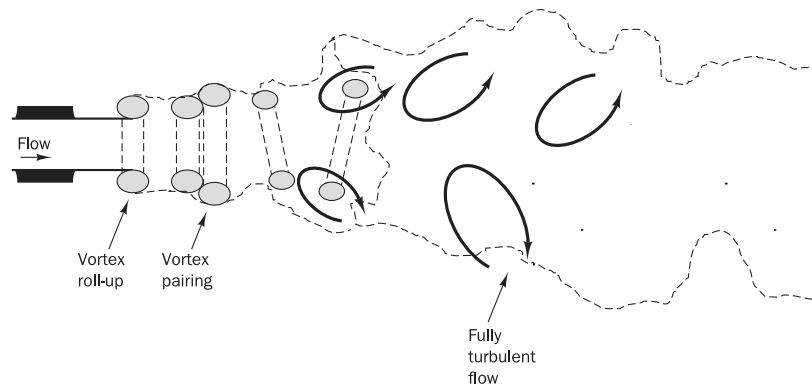
Transition to turbulence

The point where instability first occurs is always upstream of the point of transition to fully turbulent flow. The distance between the point of instability where the Reynolds number equals $Re_{x,crit}$ and the point of transition

$Re_{x,tr}$ depends on the degree of amplification of the unstable disturbances. The point of instability and the onset of the transition process can be predicted with the linear theory of hydrodynamic instability. There is, however, no comprehensive theory regarding the path leading from initial instability to fully turbulent flows. Next, we describe the main, experimentally observed, characteristics of three simple flows: jets, flat plate boundary layers and pipe flows.

Jet flow: an example of a flow with a point of inflexion

Flows which possess one or more points of inflexion amplify long-wavelength disturbances at all Reynolds numbers typically above about 10. The transition process is explained by considering the sketch of a jet flow (Figure 3.5).

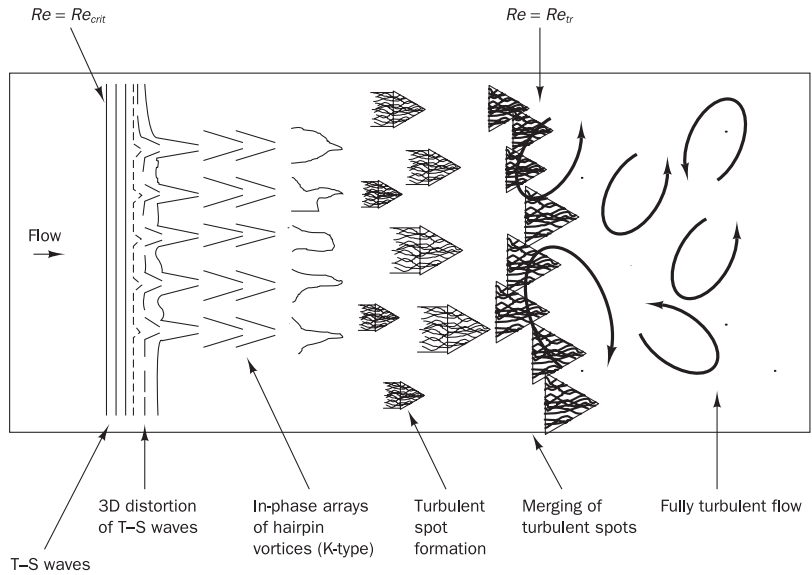
Figure 3.5 Transition in a jet flow

After the flow emerges from the orifice the laminar exit flow produces the rolling up of a vortex fairly close to the orifice. Subsequent amplification involves the formation of a single vortex of greater strength through the pairing of vortices. A short distance further downstream, three-dimensional disturbances cause the vortices to become heavily distorted and less distinct. The flow breaks down, generating a large number of small-scale eddies, and the flow undergoes rapid transition to the fully turbulent regime. Mixing layers and wakes behind bluff bodies exhibit a similar sequence of events, leading to transition and turbulent flow.

Boundary layer on a flat plate: an example of a flow without a point of inflexion

In flows with a velocity distribution without a point of inflexion viscous instability theory predicts that there is a finite region of Reynolds numbers around $Re_\delta = 1000$ (δ is the boundary layer thickness) where infinitesimal disturbances are amplified. The developing flow over a flat plate is such a flow, and the transition process has been extensively researched for this case.

The precise sequence of events is sensitive to the level of disturbance of the incoming flow. However, if the flow system creates sufficiently smooth conditions the instability of a boundary layer flow to relatively long-wavelength

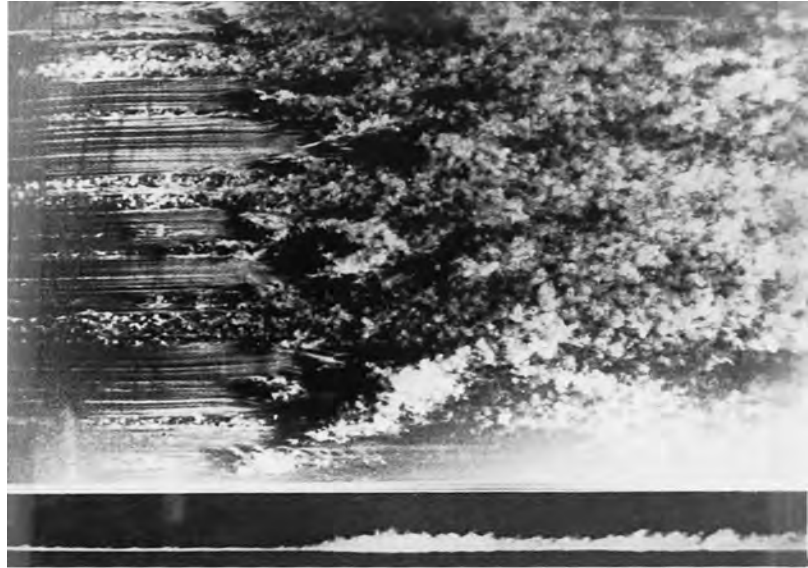
disturbances can be clearly detected. A sketch of the processes leading to transition and fully turbulent flow is given in Figure 3.6.

Figure 3.6 Plan view sketch of transition processes in boundary layer flow over a flat plate

If the incoming flow is laminar numerous experiments confirm the predictions of the theory that initial linear instability occurs around $Re_{x,crit}$ = 91 000. The unstable two-dimensional disturbances are called Tollmien–Schlichting (T–S) waves. These disturbances are amplified in the flow direction.

The subsequent development depends on the amplitude of the waves at maximum (linear) amplification. Since amplification takes place over a limited range of Reynolds numbers, it is possible that the amplified waves are attenuated further downstream and that the flow remains laminar. If the amplitude is large enough a secondary, non-linear, instability mechanism causes the Tollmien–Schlichting waves to become three-dimensional and finally evolve into hairpin Λ-vortices. In the most common mechanism of transition, so-called K-type transition, the hairpin vortices are aligned.

Above the hairpin vortices a high shear region is induced which subsequently intensifies, elongates and rolls up. Further stages of the transition process involve a cascading breakdown of the high shear layer into smaller units with frequency spectra of measurable flow parameters approaching randomness. Regions of intense and highly localised changes occur at random times and locations near the solid wall. Triangular turbulent spots burst from these locations. These turbulent spots are carried along with the flow and grow by spreading sideways, which causes increasing amounts of laminar fluid to take part in the turbulent motion.

Transition of a natural flat plate boundary layer involves the formation of turbulent spots at active sites and the subsequent merging of different turbulent spots convected downstream by the flow. This takes place at Reynolds numbers $Re_{x,tr} \approx 10^6$. Figure 3.7 is a plan view snapshot of a flat plate boundary layer that illustrates this process.

Figure 3.7 Merging of turbulent spots and transition to turbulence in a flat plate boundary layer
Source: Nakayama (1988)

Pipe flow transition

The transition in a pipe flow represents an example of a special category of flows without an inflexion point. The viscous theory of hydrodynamic stability predicts that these flows are unconditionally stable to infinitesimal disturbances at all Reynolds numbers. In practice, transition to turbulence takes place between Re (= UD/ν) 2000 and 10^5. Various details are still unclear, which illustrates the limitations of current stability theories.

The cause of the apparent failure of the theory is almost certainly the role played by distortions of the inlet velocity profile and the finite amplitude disturbances due to entry effects. Experiments show that in pipe flows, as in flat plate boundary layers, turbulent spots appear in the near-wall region. These grow, merge and subsequently fill the pipe cross-section to form turbulent slugs. In industrial pipe flows intermittent formation of turbulent slugs takes place at Reynolds numbers around 2000 giving rise to alternate turbulent and laminar regions along the length of the pipe. At Reynolds numbers above 2300 the turbulent slugs link up and the entire pipe is filled with turbulent flow.

Final comments

It is clear from the above descriptions of transition in jets, flat plate boundary layers and pipe flows that there are a number of common features in the transition processes: (i) the amplification of initially small disturbances, (ii) the development of areas with concentrated rotational structures, (iii) the formation of intense small-scale motions and finally (iv) the growth and merging of these areas of small-scale motions into fully turbulent flows.

The transition to turbulence is strongly affected by factors such as pressure gradient, disturbance levels, wall roughness and heat transfer. The discussions only apply to subsonic, incompressible flows. The appearance of

significant compressibility effects in flows at Mach numbers above about 0.7 greatly complicates the stability theory.

It should be noted that although a great deal has been learnt from simple flows, there is no comprehensive theory of transition. Advances in supercomputer technology have made it possible to simulate the events leading up to transition, including turbulent spot formation, and turbulence at modest Reynolds numbers by solving the complete, time-dependent Navier–Stokes equations for simple geometries. Kleiser and Zang (1991) gave a review which highlights very favourable agreement between experiments and their computations.

For engineering purposes the major case where the transition process influences a sizeable fraction of the flow is that of external wall boundary layer flows at intermediate Reynolds numbers. This occurs in certain turbomachines, helicopter rotors and some low-speed aircraft wings. Cebeci (1989) presented an engineering calculation method based on a combination of inviscid far field and boundary layer computations in conjunction with a linear stability analysis to identify the critical and transition Reynolds numbers. Transition is deemed to have occurred at the point where an (arbitrary) amplification factor e^9 (≈ 8000) of initial disturbances is found. The procedure, which includes a mixing length model (see section 3.6.1) for the fully turbulent part of the boundary layer, has proved very effective for aerofoil calculations, but requires a substantial amount of empirical input and therefore lacks generality.

Commercially available general-purpose CFD procedures often ignore transition entirely and classify flows as either laminar or fully turbulent. The transition region often constitutes only a very small fraction of the size of the flow domain and in those cases it is assumed that the errors made by neglecting its detailed structure are only small.

3.3 Descriptors of turbulent flow

Let us consider a single point measurement in a turbulent flow, e.g. a velocity measurement made with a hot-wire anemometer (Comte-Bellot, 1976) or a laser Doppler anemometer (Buchhave *et al*., 1979) or a local pressure measurement made with a small transducer. In Figure 3.1, we saw that the appearance of turbulence manifested itself as random fluctuations of the measured velocity component about a mean value. All other flow variables, i.e. all other velocity components, the pressure, temperature, density etc., will also exhibit this additional time-dependent behaviour. The Reynolds decomposition defines flow property φ at this point as the sum of a steady mean component Φ and a time varying fluctuating component $\varphi'(t)$ with zero mean value: hence $\varphi(t) = \Phi + \varphi'(t)$. We start with a formal definition of the time average or mean Φ and we also define the most widely used statistical descriptors of the fluctuating component φ'.

Time average or mean

The **mean** Φ of flow property φ is defined as follows:

$$\Phi = \frac{1}{\Delta t}\int_0^{\Delta t} \varphi(t)\ \mathrm{d}t \tag{3.2}$$

In theory we should take the limit of time interval Δt approaching infinity, but the process indicated by equation (3.2) gives meaningful time averages if Δt is larger than the time scale associated with the slowest variations (due to the largest eddies) of property φ. This definition of the mean of a flow property is adequate for steady mean flows. In time-dependent flows the mean of a property at time t is taken to be the average of the instantaneous values of the property over a large number of repeated identical experiments: the so-called 'ensemble average'.

The time average of the fluctuations φ' is, by definition, zero:

$$\overline{\varphi'} = \frac{1}{\Delta t}\int_0^{\Delta t} \varphi'(t)\, dt \equiv 0 \tag{3.3}$$

From now on we shall not write down the time-dependence of φ and φ' explicitly, so we write $\varphi = \Phi + \varphi'$.

The most compact description of the main characteristics of the fluctuating component of a turbulent flow variable is in terms of its statistics.

Variance, r.m.s. and turbulence kinetic energy

The descriptors used to indicate the spread of the fluctuations φ' about the mean value Φ are the **variance** and **root mean square** (r.m.s.):

$$\overline{(\varphi')^2} = \frac{1}{\Delta t}\int_0^{\Delta t} (\varphi')^2\, dt \tag{3.4a}$$

$$\varphi_{rms} = \sqrt{\overline{(\varphi')^2}} = \left[\frac{1}{\Delta t}\int_0^{\Delta t} (\varphi')^2\, dt\right]^{1/2} \tag{3.4b}$$

The r.m.s. values of the velocity components are of particular importance since they are generally most easily measured and express the average magnitude of velocity fluctuations. In section 3.5 we will come across the variances of velocity fluctuations $\overline{u'^2}$, $\overline{v'^2}$ and $\overline{w'^2}$ when we consider the time average of the Navier–Stokes equations and find that they are proportional to the momentum fluxes induced by turbulent eddies, which cause additional normal stresses experienced by fluid elements in a turbulent flow.

One-half times these variances has a further interpretation as the mean kinetic energy per unit mass contained in the respective velocity fluctuations. The total kinetic energy per unit mass k of the turbulence at a given location can be found as follows:

$$k = \frac{1}{2}\left(\overline{u'^2} + \overline{v'^2} + \overline{w'^2}\right) \tag{3.5}$$

The turbulence intensity T_i is the average r.m.s. velocity divided by a reference mean flow velocity U_{ref} and is linked to the turbulence kinetic energy k as follows:

$$T_i = \frac{(\frac{2}{3}k)^{1/2}}{U_{ref}} \tag{3.6}$$

Moments of different fluctuating variables

The variance is also called the second moment of the fluctuations. Important details of the structure of the fluctuations are contained in moments constructed from pairs of different variables. For example, consider properties $\varphi = \Phi + \varphi'$ and $\psi = \Psi + \psi'$ with $\overline{\varphi'} = \overline{\psi'} = 0$. Their **second moment** is defined as

$$\overline{\varphi'\psi'} = \frac{1}{\Delta t}\int_0^{\Delta t} \varphi'\psi'\,\mathrm{d}t \tag{3.7}$$

If velocity fluctuations in different directions were independent random fluctuations, then the values of the second moments of the velocity components $\overline{u'v'}$, $\overline{u'w'}$ and $\overline{v'w'}$ would be equal to zero. However, as we have seen, turbulence is associated with the appearance of vortical flow structures and the induced velocity components are chaotic, but not independent, so in turn the second moments are non-zero. In section 3.5 we will come across $\overline{u'v'}$, $\overline{u'w'}$ and $\overline{v'w'}$ again in the time-average of the Navier–Stokes equations. They represent turbulent momentum fluxes that are closely linked with the additional shear stresses experienced by fluid elements in turbulent flows. Pressure–velocity moments, $\overline{p'u'}$, $\overline{p'v'}$ etc., play a role in the diffusion of turbulent energy.

Higher-order moments

Additional information relating to the distribution of the fluctuations can be obtained from higher-order moments. In particular, the third and fourth moments are related to the skewness (asymmetry) and kurtosis (peakedness), respectively:

$$\overline{(\varphi')^3} = \frac{1}{\Delta t}\int_0^{\Delta t} (\varphi')^3\,\mathrm{d}t \tag{3.8}$$

$$\overline{(\varphi')^4} = \frac{1}{\Delta t}\int_0^{\Delta t} (\varphi')^4\,\mathrm{d}t \tag{3.9}$$

Correlation functions – time and space

More detailed information relating to the structure of the fluctuations can be obtained by studying the relationship between the fluctuations at different times. The **autocorrelation** function $R_{\varphi'\varphi'}(\tau)$ is defined as

$$R_{\varphi'\varphi'}(\tau) = \overline{\varphi'(t)\varphi'(t+\tau)} = \frac{1}{\Delta t}\int_0^{\Delta t} \varphi'(t)\varphi'(t+\tau)\,\mathrm{d}t \tag{3.10}$$

Similarly, it is possible to define a further **autocorrelation** function $R_{\varphi'\varphi'}(\xi)$ based on two measurements shifted by a certain distance in space:

$$R_{\varphi'\varphi'}(\xi) = \overline{\varphi'(\mathbf{x},t)\varphi'(\mathbf{x}+\xi,t)} = \frac{1}{\Delta t}\int_t^{t+\Delta t} \varphi'(\mathbf{x},t')\varphi'(\mathbf{x}+\xi,t')\,\mathrm{d}t' \tag{3.11}$$

When time shift τ (or displacement ξ) is zero the value of the autocorrelation function $R_{\varphi'\varphi'}(0)$ (or $R_{\varphi'\varphi'}(\underline{\mathbf{0}})$) just corresponds to the variance $\overline{\varphi'^2}$ and will have its largest possible value, because the two contributions are perfectly correlated. Since the behaviour of the fluctuations φ' is chaotic in a turbulent flow, we expect that the fluctuations become increasingly decorrelated as $\tau \to \infty$ (or $|\xi|\to\infty$), so values of the time or space autocorrelation functions will decrease to zero. The eddies at the root of turbulence cause a certain degree of local structure in the flow, so there will be correlation between the values of φ' at time t and a short time later or at a given location $\mathbf{x}$ and a small distance away. The decorrelation process will take place gradually over the lifetime (or size scale) of a typical eddy. The integral time and length scale, which represent concrete measures of the average period or size of a turbulent eddy, can be computed from integrals of the autocorrelation function $R_{\varphi'\varphi'}(\tau)$ with respect to time shift τ or $R_{\varphi'\varphi'}(\xi)$ with respect to distance in the direction of one of the components of displacement vector ξ.

By analogy it is also possible to define **cross-correlation functions** $R_{\varphi'\psi'}(\tau)$ with respect to time shift τ or $R_{\varphi'\psi'}(\xi)$ between pairs of different fluctuations by replacing the second φ' by ψ' in equations (3.10) and (3.11).

Probability density function

Finally, we mention the **probability density function** $P(\varphi^*)$, which is related to the fraction of time that a fluctuating signal spends between φ^* and $\varphi^* + \mathrm{d}\varphi$. This is defined in terms of a probability as follows:

$$P(\varphi^*)\mathrm{d}\varphi^* = \mathrm{Prob}(\varphi^* < \varphi < \varphi^* + \mathrm{d}\varphi^*) \qquad (3.12)$$

The average, variance and higher moments of the variable and its fluctuations are related to the probability density function as follows:

$$\overline{\varphi} = \int_{-\infty}^{\infty} \varphi P(\varphi)\mathrm{d}\varphi \qquad (3.13a)$$

$$\overline{(\varphi')^n} = \int_{-\infty}^{\infty} (\varphi')^n P(\varphi')\mathrm{d}\varphi' \qquad (3.13b)$$

In equation (3.13b) we can use $n = 2$ to obtain the variance of φ' and $n = 3, 4$ . . . for higher-order moments. Probability density functions are used extensively in the modelling of combustion and we come across them again in Chapter 12.

3.4 Characteristics of simple turbulent flows

Most of the theory of turbulent flow was initially developed by careful examination of the turbulence structure of thin shear layers. In such flows large velocity changes are concentrated in thin regions. Expressed more formally, the rates of change of flow variables in the (x-)direction of the flow are negligible compared with the rates of change in the cross-stream (y-)direction ($\partial\varphi/\partial x \ll \partial\varphi/\partial y$). Furthermore, the cross-stream width δ of the region over which changes take place is always small compared with any length scale L in the flow direction ($\delta/L \ll 1$). In the context of this brief introduction we review the characteristics of some simple two-dimensional incompressible

turbulent flows with constant imposed pressure. The following flows will be considered here:

Free turbulent flows
- mixing layer
- jet
- wake

Boundary layers near solid walls
- flat plate boundary layer
- pipe flow

We review data for the mean velocity distribution $U = U(y)$ and the pertinent second moments $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $\overline{u'v'}$.

3.4.1 Free turbulent flows

Among the simplest flows of significant engineering importance are those in the category of free turbulent flows: mixing layers, jets and wakes. A mixing layer forms at the interface of two regions: one with fast and the other with slow moving fluid. In a jet a region of high-speed flow is completely surrounded by stationary fluid. A wake is formed behind an object in a flow, so here a slow moving region is surrounded by fast moving fluid. Figure 3.8 is a sketch of the development of the mean velocity distribution in the streamwise direction for these free turbulent flows.

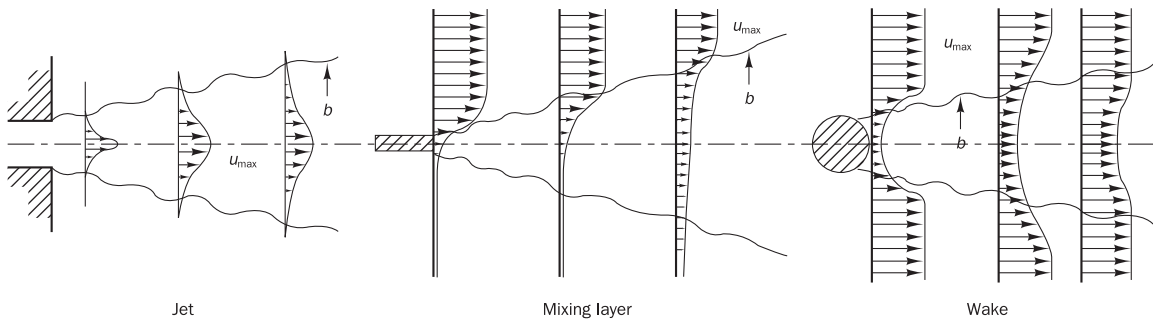



Figure 3.8 Free turbulent flows

It is clear that velocity changes across an initially thin layer are important in all three flows. Transition to turbulence occurs after a very short distance in the flow direction from the point where the different streams initially meet; the turbulence causes vigorous mixing of adjacent fluid layers and rapid widening of the region across which the velocity changes take place.

Figure 3.9 shows a visualisation of a jet flow. It is immediately clear that the turbulent part of the flow contains a wide range of length scales. Large eddies with a size comparable to the width across the flow are occurring alongside eddies of very small size.

The visualisation correctly suggests that the flow inside the jet region is fully turbulent, but the flow in the outer region far away from the jet is smooth and largely unaffected by the turbulence. The position of the edge of the turbulent zone is determined by the (time-dependent) passage of individual large eddies. Close to the edge these will occasionally penetrate into

Figure 3.9 Visualisation of a jet flow
Source: Van Dyke (1982)

the surrounding region. During the resulting bursts of turbulent activity in the outer region – called **intermittency** – fluid from the surroundings is drawn into the turbulent zone. This process is termed **entrainment** and is the main cause of the spreading of turbulent flows (including wall boundary layers) in the flow direction.

Initially fast moving jet fluid will lose momentum to speed up the stationary surrounding fluid. Due to the entrainment of surrounding fluid the velocity gradients decrease in magnitude in the flow direction. This causes the decrease of the mean speed of the jet at its centreline. Similarly the difference between the speed of the wake fluid and its fast moving surroundings will decrease in the flow direction. In mixing layers the width of the layer containing the velocity change continues to increase in the flow direction but the overall velocity difference between the two outer regions is unaltered.

Experimental observations of many such turbulent flows show that after a certain distance their structure becomes independent of the exact nature of the flow source. Only the local environment appears to control the turbulence in the flow. The appropriate length scale is the cross-stream layer width (or half width) b. We find that if y is the distance in the cross-stream direction

$$\boxed{\frac{U - U_{\min}}{U_{\max} - U_{\min}} = f\left(\frac{y}{b}\right)} \quad \boxed{\frac{U}{U_{\max}} = g\left(\frac{y}{b}\right)} \quad \boxed{\frac{U_{\max} - U}{U_{\max} - U_{\min}} = h\left(\frac{y}{b}\right)} \tag{3.14}$$

for mixing layers for jets for wakes

In these formulae $U_{\max}$ and $U_{\min}$ represent the maximum and minimum mean velocity at a distance x downstream of the source (see Figure 3.8). Hence, if these local mean velocity scales are chosen and x is large enough, the functions f, g and h are independent of distance x in the flow direction. Such flows are called **self-preserving**.

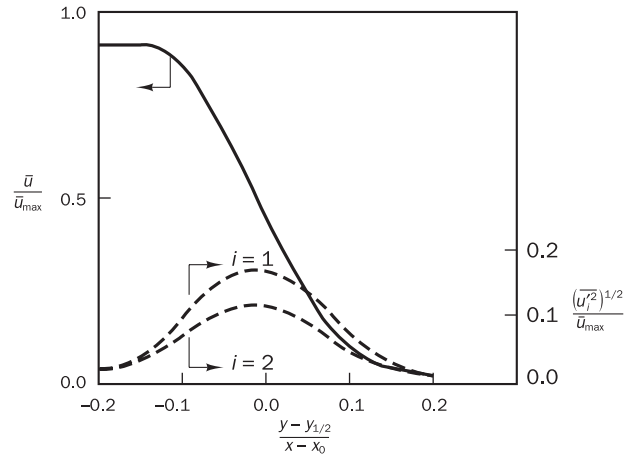
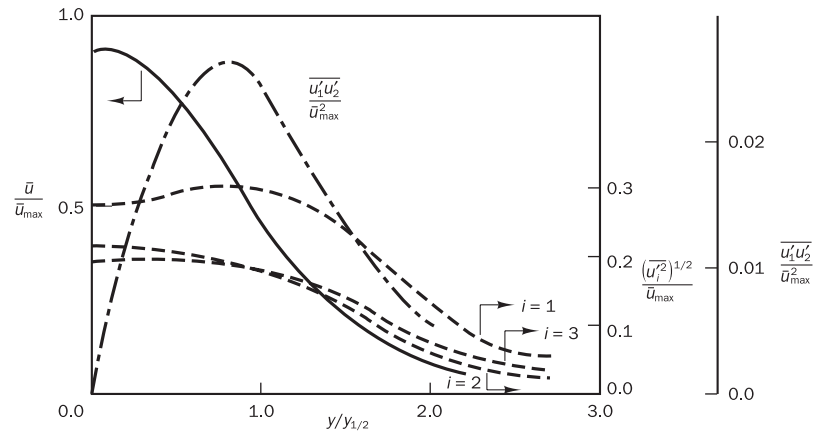
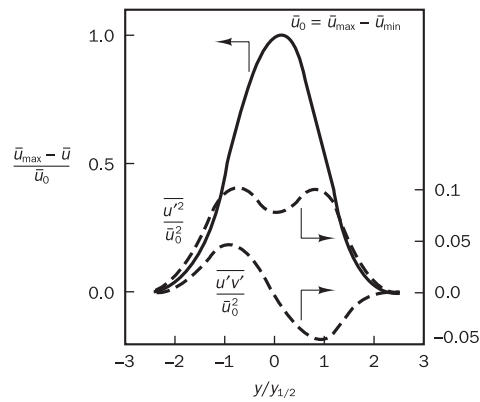
The turbulence structure also reaches a self-preserving state, albeit after a greater distance from the flow source than the mean velocity. Then

$$\boxed{\frac{\overline{u'^2}}{U_{ref}^2} = f_1\left(\frac{y}{b}\right)} \quad \boxed{\frac{\overline{v'^2}}{U_{ref}^2} = f_2\left(\frac{y}{b}\right)} \quad \boxed{\frac{\overline{w'^2}}{U_{ref}^2} = f_3\left(\frac{y}{b}\right)} \quad \boxed{\frac{\overline{u'v'}}{U_{ref}^2} = f_4\left(\frac{y}{b}\right)} \tag{3.15}$$

The velocity scale U_{ref} is, as above, $(U_{\max} - U_{\min})$ for a mixing layer and wakes and $U_{\max}$ for jets. The precise form of functions f, g, h and f_i varies from flow to flow. Figure 3.10 gives mean velocity and turbulence data for a mixing layer (Champagne *et al*., 1976), a jet (Gutmark and Wygnanski, 1976) and a wake flow (Wygnanski *et al*., 1986).

The largest values of $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $-\overline{u'v'}$ are found in the region where the mean velocity gradient $\partial U/\partial y$ is largest, highlighting the intimate connection between turbulence production and sheared mean flows. In the flows shown above the component u' gives the largest of the normal stresses; its r.m.s. value has a maximum of 15–40% of the local maximum mean flow velocity. The fact that the fluctuating velocities are not equal implies an **anisotropic** structure of the turbulence.

As $|y/b|$ increases above unity the mean velocity gradients and the velocity fluctuations all tend to zero. It should also be noted that the turbulence

Figure 3.10 Distribution of mean velocity and second moments $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $-\overline{u'v'}$ for incompressible mixing layer, jet and wake

properties become more **isotropic**. The absence of shear means that turbulence cannot be sustained in this region.

The mean velocity gradient is also zero at the centreline of jets and wakes and hence no turbulence is produced here. Nevertheless, the values of $\overline{u'^2}$, $\overline{v'^2}$ and $\overline{w'^2}$ do not decrease very much because vigorous eddy mixing transports turbulent fluid from nearby regions of high turbulence production towards and across the centreline. The value of $-\overline{u'v'}$ has to become zero at the centreline of jet and wake flows since it must change sign here by symmetry.

3.4.2 Flat plate boundary layer and pipe flow

Next we will examine the characteristics of two turbulent flows near solid walls. Due to the presence of the solid boundary, the flow behaviour and turbulence structure are considerably different from free turbulent flows. Dimensional analysis has greatly assisted in correlating the experimental data. In turbulent thin shear layer flows a Reynolds number based on a length scale L in the flow direction (or pipe radius) Re_L is always very large (e.g. $U = 1$ m/s, $L = 0.1$ m and $\nu = 10^{-6}$ m^2/s gives $Re_L = 10^5$). This implies that the inertia forces are overwhelmingly larger than the viscous forces at these scales.

If we form a Reynolds number based on a distance y away from the wall ($Re_y = Uy/\nu$) we see that if the value of y is of the order of L the above argument holds. Inertia forces dominate in the flow far away from the wall. As y is decreased to zero, however, a Reynolds number based on y will also decrease to zero. Just before y reaches zero there will be a range of values of y for which Re_y is of the order of 1. At this distance from the wall and closer the viscous forces will be equal in order of magnitude to inertia forces or larger. To sum up, in flows along solid boundaries there is usually a substantial region of inertia-dominated flow far away from the wall and a thin layer within which viscous effects are important.

Close to the wall the flow is influenced by viscous effects and does not depend on free stream parameters. The mean flow velocity only depends on the distance y from the wall, fluid density ρ and viscosity μ and the wall shear stress τ_w. So

$$U = f(y, \rho, \mu, \tau_w)$$

Dimensional analysis shows that

$$\boxed{u^+ = \frac{U}{u_\tau} = f\left(\frac{\rho u_\tau y}{\mu}\right) = f(y^+)} \tag{3.16}$$

Formula (3.16) is called the **law of the wall** and contains the definitions of two important dimensionless groups, u^+ and y^+. Note that the appropriate velocity scale is $u_\tau = \sqrt{\tau_w/\rho}$, the so-called friction velocity.

Far away from the wall we expect the velocity at a point to be influenced by the retarding effect of the wall through the value of the wall shear stress, but not by the viscosity itself. The length scale appropriate to this region is the boundary layer thickness δ. In this region we have

$$U = g(y, \delta, \rho, \tau_w)$$

Dimensional analysis yields

$$u^+ = \frac{U}{u_\tau} = g\left(\frac{y}{\delta}\right)$$

The most useful form emerges if we view the wall shear stress as the cause of a velocity deficit $U_{max} - U$ which decreases the closer we get to the edge of the boundary layer or the pipe centreline. Thus

$$\boxed{\frac{U_{max} - U}{u_\tau} = g\left(\frac{y}{\delta}\right)} \tag{3.17}$$

This formula is called the **velocity-defect law**.

Linear or viscous sub-layer – the fluid layer in contact with a smooth wall

At the solid surface the fluid is stationary. Turbulent eddying motions must also stop very close to the wall and the behaviour of the fluid closest to the wall is dominated by viscous effects. This **viscous sub-layer** is in practice extremely thin ($y^+ < 5$) and we may assume that the shear stress is approximately constant and equal to the wall shear stress τ_w throughout the layer. Thus

$$\tau(y) = \mu\frac{\partial U}{\partial y} \cong \tau_w$$

After integration with respect to y and application of boundary condition $U = 0$ if $y = 0$, we obtain a linear relationship between the mean velocity and the distance from the wall

$$U = \frac{\tau_w y}{\mu}$$

After some simple algebra and making use of the definitions of u^+ and y^+ this leads to

$$\boxed{u^+ = y^+} \tag{3.18}$$

Because of the linear relationship between velocity and distance from the wall the fluid layer adjacent to the wall is also known as the **linear sub-layer**.

Log-law layer – the turbulent region close to a smooth wall

Outside the viscous sublayer ($30 < y^+ < 500$) a region exists where viscous and turbulent effects are both important. The shear stress τ varies slowly with distance from the wall, and within this inner region it is assumed to be constant and equal to the wall shear stress. One further assumption regarding the length scale of turbulence (mixing length $\ell_m = \kappa y$, see section 3.7.1 and Schlichting, 1979) allows us to derive a functional relationship between u^+ and y^+ that is dimensionally correct:

$$\boxed{u^+ = \frac{1}{\kappa}\ln(y^+) + B = \frac{1}{\kappa}\ln(Ey^+)} \tag{3.19}$$

Numerical values for the constants are found from measurements. We find von Karman's constant $\kappa \approx 0.4$ and the additive constant $B \approx 5.5$ (or $E \approx 9.8$) for smooth walls; wall roughness causes a decrease in the value of B. The values of κ and B are universal constants valid for all turbulent flows past smooth walls at high Reynolds number. Because of the logarithmic relationship between u^+ and y^+, formula (3.18) is often called the **log-law**, and the layer where y^+ takes values between 30 and 500 the **log-law layer**.

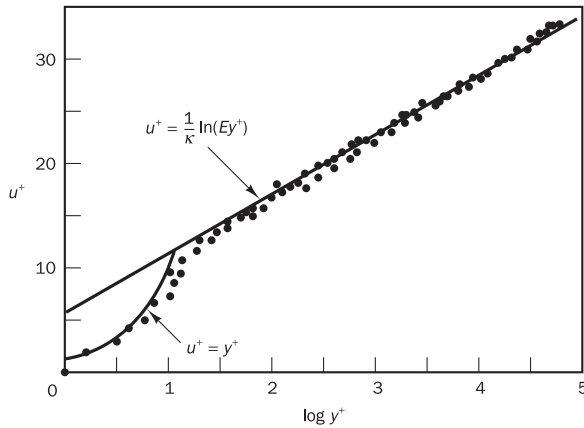
Outer layer – the inertia-dominated region far from the wall

Experimental measurements show that the log-law is valid in the region $0.02 < y/\delta < 0.2$. For larger values of y the velocity-defect law (3.17) provides the correct form. In the overlap region the log-law and velocity-defect law have to be equal. Tennekes and Lumley (1972) show that a matched overlap is obtained by assuming the following logarithmic form:

$$\frac{U_{max} - U}{u_\tau} = -\frac{1}{\kappa} \ln\left(\frac{y}{\delta}\right) + A \tag{3.20}$$

where A is a constant. This velocity-defect law is often called the **law of the wake**.

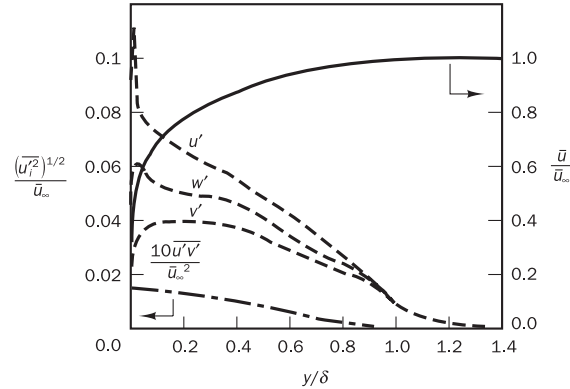
Figure 3.11 from Schlichting (1979) shows the close agreement between theoretical equations (3.18) and (3.19) in their respective areas of validity and experimental data.

Figure 3.11 Velocity distribution near a solid wall *Source*: Schlichting, H. (1979) *Boundary Layer Theory*, 7th edn, reproduced with permission of The McGraw-Hill Companies

The turbulent boundary layer adjacent to a solid surface is composed of two regions:

- The inner region: 10–20% of the total thickness of the wall layer; the shear stress is (almost) constant and equal to the wall shear stress τ_w. Within this region there are three zones. In order of increasing distance from the wall we have:
  - the linear sub-layer: viscous stresses dominate the flow adjacent to surface
  - the buffer layer: viscous and turbulent stresses are of similar magnitude
  - the log-law layer: turbulent (Reynolds) stresses dominate.

- The outer region or law-of-the-wake layer: inertia-dominated core flow far from wall; free from direct viscous effects.

Figure 3.12 shows the mean velocity and turbulence property distribution data for a flat plate boundary layer with a constant imposed pressure (Klebanoff, 1955).

Figure 3.12 Distribution of mean velocity and second moments $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $-\overline{u'v'}$ for flat plate boundary layer

The mean velocity is at a maximum far away from the wall and sharply decreases in the region $y/\delta \leq 0.2$ due to the no-slip condition. High values of $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $-\overline{u'v'}$ are found adjacent to the wall where the large mean velocity gradients ensure that **turbulence production** is high. The eddying motions and associated velocity fluctuations are, however, also subject to the no-slip condition at the wall. Therefore all turbulent stresses decrease sharply to zero in this region. The turbulence is **strongly anisotropic** near the wall since the production process mainly creates component $\overline{u'^2}$. This is borne out by the fact that this is the largest of the mean-squared fluctuations in Figure 3.12.

In the case of the flat plate boundary layer the turbulence properties asymptotically tend towards zero as y/δ increases above a value of 0.8. The r.m.s. values of all fluctuating velocities become almost equal here, indicating that the turbulence structure becomes more isotropic far away from the wall. In pipe flows, on the other hand, the eddying motions transport turbulence across the centreline from areas of high production. Therefore, the r.m.s. fluctuations remain comparatively large in the centre of a pipe. By symmetry the value of $-\overline{u'v'}$ has to go to zero and change sign at the centreline.

This multi-layer structure is a universal feature of turbulent boundary layers near solid surfaces. Monin and Yaglom (1971) plotted data from Klebanoff (1955) and Laufer (1952) in the near-wall region and found not only the universal mean velocity distribution but also that data for second moments $\overline{u'^2}$, $\overline{v'^2}$, $\overline{w'^2}$ and $-\overline{u'v'}$ for flat plates and pipes collapse onto a single curve if they are non-dimensionalised with the correct velocity scale u_τ. Between these distinct layers there are intermediate zones which ensure that the various distributions merge smoothly. Interested readers may find further details including formulae which cover the whole inner region and the log-law/law-of-the-wake layer in Schlichting (1979) and White (1991).

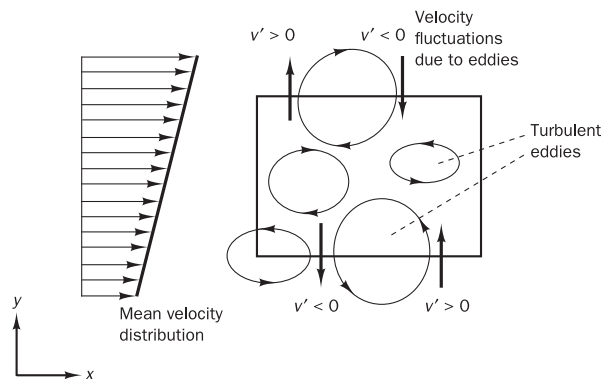
3.4.3 Summary

Our review of the characteristics of a number of two-dimensional turbulent flows revealed many common features. Turbulence is generated and maintained by shear in the mean flow. Where shear is large the magnitudes of turbulence quantities such as the r.m.s. velocity fluctuations are high and their distribution is anisotropic with higher levels of fluctuations in the mean flow direction. Without shear, or an alternative agency to maintain it, turbulence decays and becomes more isotropic in the process. In spite of these common features, it was clear that, even in these relatively simple thin shear layers, the details of the turbulence structure are very much dependent on the flow itself. In regions close to solid walls the structure is dominated by shear due to wall friction and damping of turbulent velocity fluctuations perpendicular to the boundary. This results in a complex flow structure characterised by rapid changes in the mean and fluctuating velocity components concentrated within a very narrow region in the immediately vicinity of the wall. Since most engineering flows contain solid boundaries, the turbulence structure generated by them will be very geometry dependent. Engineering flow calculations must include sufficiently accurate and general descriptions of the turbulence that capture all the above effects and further interactions of turbulence and body forces.

3.5 The effect of turbulent fluctuations on properties of the mean flow

In this section we derive the flow equations governing the time-averaged properties of a turbulent flow, but before we do this we briefly examine the physical basis of the effects resulting from the appearance of turbulent fluctuations.

In Figure 3.13 we consider a control volume in a two-dimensional turbulent shear flow parallel to the x-axis with a mean velocity gradient in the y-direction. The presence of vortical eddy motions creates strong mixing. Random currents that are associated with the passage of eddies near the boundaries of the control volume transport fluid across its boundaries. These recirculating fluid motions cannot create or destroy mass, but fluid parcels transported by the eddies will carry momentum and energy into and out of the control volume. Figure 3.13 shows that, because of the existence of the velocity gradient, fluctuations with a negative y-velocity will generally bring

Figure 3.13 Control volume within a two-dimensional turbulent shear flow

fluid parcels with a higher x-momentum into the control volume across its top boundary and will also transport control volume fluid to a region of slower moving fluid across the bottom boundary. Similarly, positive y-velocity fluctuations will – on average – transport slower moving fluid into regions of higher velocity. The net result is **momentum exchange** due to convective transport by the eddies, which causes the faster moving fluid layers to be decelerated and the slower moving layers to be accelerated. Consequently, the fluid layers experience additional turbulent shear stresses, which are known as the **Reynolds stresses**. In the presence of temperature or concentration gradients the eddy motions will also generate **turbulent heat or species concentration fluxes** across the control volume boundaries. This discussion suggests that the equations for momentum and energy should be affected by the appearance of fluctuations.

Reynolds-averaged Navier–Stokes equations for incompressible flow

Next we examine the consequences of turbulent fluctuations for the mean flow equations for an incompressible flow with constant viscosity. These assumptions considerably simplify the algebra involved without detracting from the main messages. We begin by summarising the rules which govern time averages of fluctuating properties $\varphi = \Phi + \varphi'$ and $\psi = \Psi + \psi'$ and their summation, derivatives and integrals:

$$\overline{\varphi'} = \overline{\psi'} = 0 \quad \overline{\Phi} = \Phi \quad \overline{\frac{\partial \varphi}{\partial s}} = \frac{\partial \Phi}{\partial s} \quad \overline{\int \varphi \mathrm{d}s} = \int \Phi \mathrm{d}s \tag{3.21}$$

$$\overline{\varphi + \psi} = \Phi + \Psi \quad \overline{\varphi\psi} = \Phi\Psi + \overline{\varphi'\psi'} \quad \overline{\varphi\Psi} = \Phi\Psi \quad \overline{\varphi'\Psi} = 0$$

These relationships can be easily verified by application of (3.2) and (3.3), noting that the time-averaging operation is itself an integration. Thus, the order of time averaging and summation, further integration and/or differentiation can be swapped or commuted, so this is called the **commutative property**.

Since div and grad are both differentiations, the above rules can be extended to a fluctuating vector quantity $\mathbf{a} = \mathbf{A} + \mathbf{a}'$ and its combinations with a fluctuating scalar $\varphi = \Phi + \varphi'$:

$$\overline{\operatorname{div} \mathbf{a}} = \operatorname{div} \mathbf{A}; \ \overline{\operatorname{div}(\varphi \mathbf{a})} = \operatorname{div}(\overline{\varphi \mathbf{a}}) = \operatorname{div}(\Phi \mathbf{A}) + \operatorname{div}(\overline{\varphi' \mathbf{a}'});$$
$$\overline{\operatorname{div} \operatorname{grad} \varphi} = \operatorname{div} \operatorname{grad} \Phi \tag{3.22}$$

To start with we consider the instantaneous continuity and Navier–Stokes equations in a Cartesian co-ordinate system so that the velocity vector $\mathbf{u}$ has x-component u, y-component v and z-component w:

$$\operatorname{div} \mathbf{u} = 0 \tag{3.23}$$

$$\frac{\partial u}{\partial t} + \operatorname{div}(u\mathbf{u}) = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu \operatorname{div}(\operatorname{grad}(u)) \tag{3.24a}$$

$$\frac{\partial v}{\partial t} + \operatorname{div}(v\mathbf{u}) = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu \operatorname{div}(\operatorname{grad}(v)) \tag{3.24b}$$

$$\frac{\partial w}{\partial t} + \operatorname{div}(w\mathbf{u}) = -\frac{1}{\rho}\frac{\partial p}{\partial z} + \nu \operatorname{div}(\operatorname{grad}(w)) \tag{3.24c}$$

This system of equations governs every turbulent flow, but we investigate the effects of fluctuations on the mean flow using the Reynolds decomposition in equations (3.23) and (3.24a–c) and replace the flow variables **u** (hence also u, v and w) and p by the sum of a mean and fluctuating component. Thus

$$\mathbf{u} = \mathbf{U} + \mathbf{u}' \quad u = U + u' \quad v = V + v' \quad w = W + w' \quad p = P + p'$$

Then the time average is taken, applying the rules stated in (3.21)–(3.22). Considering the continuity equation (3.23), first we note that $\overline{\text{div}\,\mathbf{u}} = \text{div}\,\mathbf{U}$. This yields the **continuity equation for the mean flow**:

$$\boxed{\text{div}\,\mathbf{U} = 0} \tag{3.25}$$

A similar process is now carried out on the x-momentum equation (3.24a). The time averages of the individual terms in this equation can be written as follows:

$$\overline{\frac{\partial u}{\partial t}} = \frac{\partial U}{\partial t} \qquad \overline{\text{div}(u\mathbf{u})} = \text{div}(U\mathbf{U}) + \text{div}(\overline{u'\mathbf{u}'})$$

$$\overline{-\frac{1}{\rho}\frac{\partial p}{\partial x}} = -\frac{1}{\rho}\frac{\partial P}{\partial x} \qquad \overline{\nu\,\text{div}(\text{grad}(u))} = \nu\,\text{div}(\text{grad}(U))$$

Substitution of these results gives the **time-average x-momentum equation**

$$\frac{\partial U}{\partial t} + \text{div}(U\mathbf{U}) + \text{div}(\overline{u'\mathbf{u}'}) = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\,\text{div}(\text{grad}(U)) \tag{3.26a}$$

(I) (II) (III) (IV) (V)

Repetition of this process on equations (3.24b) and (3.24c) yields the **time-average y- and z-momentum equations**:

$$\frac{\partial V}{\partial t} + \text{div}(V\mathbf{U}) + \text{div}(\overline{v'\mathbf{u}'}) = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \nu\,\text{div}(\text{grad}(V)) \tag{3.26b}$$

(I) (II) (III) (IV) (V)

$$\frac{\partial W}{\partial t} + \text{div}(W\mathbf{U}) + \text{div}(\overline{w'\mathbf{u}'}) = -\frac{1}{\rho}\frac{\partial P}{\partial z} + \nu\,\text{div}(\text{grad}(W)) \tag{3.26c}$$

(I) (II) (III) (IV) (V)

It is important to note that the terms (I), (II), (IV) and (V) in (3.26a–c) also appear in the instantaneous equations (3.24a–c), but the process of time averaging has introduced new terms (III) in the resulting time-average momentum equations. The terms involve products of fluctuating velocities and are associated with convective momentum transfer due to turbulent eddies. It is customary to place these terms on the right hand side of equations (3.26a–c) to reflect their role as additional turbulent stresses on the mean velocity components U, V and W:

$$\boxed{\begin{aligned}\frac{\partial U}{\partial t} + \text{div}(U\mathbf{U}) &= -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\,\text{div}(\text{grad}(U)) \\ &\quad + \frac{1}{\rho}\left[\frac{\partial(-\rho\overline{u'^2})}{\partial x} + \frac{\partial(-\rho\overline{u'v'})}{\partial y} + \frac{\partial(-\rho\overline{u'w'})}{\partial z}\right]\end{aligned}} \tag{3.27a}$$

$$\frac{\partial V}{\partial t} + \mathrm{div}(V\mathbf{U}) = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \nu\,\mathrm{div}(\mathrm{grad}(V)) + \frac{1}{\rho}\left[\frac{\partial(-\rho\overline{u'v'})}{\partial x} + \frac{\partial(-\rho\overline{v'^2})}{\partial y} + \frac{\partial(-\rho\overline{v'w'})}{\partial z}\right] \tag{3.27b}$$

$$\frac{\partial W}{\partial t} + \mathrm{div}(W\mathbf{U}) = -\frac{1}{\rho}\frac{\partial P}{\partial z} + \nu\,\mathrm{div}(\mathrm{grad}(W)) + \frac{1}{\rho}\left[\frac{\partial(-\rho\overline{u'w'})}{\partial x} + \frac{\partial(-\rho\overline{v'w'})}{\partial y} + \frac{\partial(-\rho\overline{w'^2})}{\partial z}\right] \tag{3.27c}$$

The extra stress terms have been written out in longhand to clarify their structure. They result from six additional stresses: three normal stresses

$$\tau_{xx} = -\rho\overline{u'^2} \quad \tau_{yy} = -\rho\overline{v'^2} \quad \tau_{zz} = -\rho\overline{w'^2} \tag{3.28a}$$

and three shear stresses

$$\tau_{xy} = \tau_{yx} = -\rho\overline{u'v'} \quad \tau_{xz} = \tau_{zx} = -\rho\overline{u'w'} \quad \tau_{yz} = \tau_{zy} = -\rho\overline{v'w'} \tag{3.28b}$$

These extra turbulent stresses are called the **Reynolds stresses**. The normal stresses involve the respective variances of the x-, y- and z-velocity fluctuations. They are always non-zero because they contain squared velocity fluctuations. The shear stresses contain second moments associated with correlations between different velocity components. As was stated earlier, if two fluctuating velocity components, e.g. u' and v', were independent random fluctuations the time average $\overline{u'v'}$ would be zero. However, the correlation between pairs of different velocity components due to the structure of the vortical eddies ensures that the turbulent shear stresses are also non-zero and usually very large compared with the viscous stresses in a turbulent flow. The equation set (3.25) and (3.27a–c) is called the **Reynolds-averaged Navier–Stokes equations**.

Similar extra turbulent transport terms arise when we derive a transport equation for an arbitrary scalar quantity, e.g. temperature. The **time-average transport equation for scalar φ** is

$$\frac{\partial \Phi}{\partial t} + \mathrm{div}(\Phi\mathbf{U}) = \frac{1}{\rho}\,\mathrm{div}(\Gamma_\Phi\,\mathrm{grad}\,\Phi) + \left[-\frac{\partial\overline{u'\varphi'}}{\partial x} - \frac{\partial\overline{v'\varphi'}}{\partial y} - \frac{\partial\overline{w'\varphi'}}{\partial z}\right] + S_\Phi \tag{3.29}$$

So far we have assumed that the fluid density is constant, but in practical flows the mean density may vary and the instantaneous density always exhibits turbulent fluctuations. Bradshaw *et al.* (1981) stated that small density fluctuations do not appear to affect the flow significantly. If r.m.s. velocity fluctuations are of the order of 5% of the mean speed they show that density fluctuations are unimportant up to Mach numbers around 3 to 5. In free turbulent flows we have seen in section 3.4 that velocity fluctuations can easily reach values around 20% of the mean velocity. In such circumstances

density fluctuations start to affect the turbulence around Mach numbers of 1. To summarise the results of the current section we quote, without proof, in Table 3.1, the density-weighted averaged (or Favre-averaged; see Anderson *et al.*, 1984) form of the mean flow equations for compressible turbulent flows where effects of density fluctuations are negligible, but mean density variations are not. This form is widely used in commercial CFD packages. The symbol $\tilde{U}$ stands for the Favre-averaged velocity.

Table 3.1 Turbulent flow equations for compressible flows

Continuity $\frac{\partial \bar{\rho}}{\partial t} + \text{div}(\bar{\rho}\tilde{\mathbf{U}}) = 0$ (3.30)

Reynolds equations

$$\frac{\partial(\bar{\rho}\tilde{U})}{\partial t} + \text{div}(\bar{\rho}\tilde{U}\tilde{\mathbf{U}}) = -\frac{\partial \bar{P}}{\partial x} + \text{div}(\mu \text{ grad } \tilde{U}) + \left[-\frac{\partial(\overline{\bar{\rho}u'^2})}{\partial x} - \frac{\partial(\overline{\bar{\rho}u'v'})}{\partial y} - \frac{\partial(\overline{\bar{\rho}u'w'})}{\partial z} \right] + S_{Mx} \tag{3.31a}$$

$$\frac{\partial(\bar{\rho}\tilde{V})}{\partial t} + \text{div}(\bar{\rho}\tilde{V}\tilde{\mathbf{U}}) = -\frac{\partial \bar{P}}{\partial y} + \text{div}(\mu \text{ grad } \tilde{V}) + \left[-\frac{\partial(\overline{\bar{\rho}u'v'})}{\partial x} - \frac{\partial(\overline{\bar{\rho}v'^2})}{\partial y} - \frac{\partial(\overline{\bar{\rho}v'w'})}{\partial z} \right] + S_{My} \tag{3.31b}$$

$$\frac{\partial(\bar{\rho}\tilde{W})}{\partial t} + \text{div}(\bar{\rho}\tilde{W}\tilde{\mathbf{U}}) = -\frac{\partial \bar{P}}{\partial z} + \text{div}(\mu \text{ grad } \tilde{W}) + \left[-\frac{\partial(\overline{\bar{\rho}u'w'})}{\partial x} - \frac{\partial(\overline{\bar{\rho}v'w'})}{\partial y} - \frac{\partial(\overline{\bar{\rho}w'^2})}{\partial z} \right] + S_{Mz} \tag{3.31c}$$

Scalar transport equation

$$\frac{\partial(\bar{\rho}\tilde{\Phi})}{\partial t} + \text{div}(\bar{\rho}\tilde{\Phi}\tilde{\mathbf{U}}) = \text{div}(\Gamma_\Phi \text{ grad } \tilde{\Phi}) + \left[-\frac{\partial(\overline{\bar{\rho}u'\varphi'})}{\partial x} - \frac{\partial(\overline{\bar{\rho}v'\varphi'})}{\partial y} - \frac{\partial(\overline{\bar{\rho}w'\varphi'})}{\partial z} \right] + S_\Phi \tag{3.32}$$

where the overbar indicates a time-averaged variable and the tilde indicates a density-weighted or Favre-averaged variable

3.6 Turbulent flow calculations

Turbulence causes the appearance in the flow of eddies with a wide range of length and time scales that interact in a dynamically complex way. Given the importance of the avoidance or promotion of turbulence in engineering applications, it is no surprise that a substantial amount of research effort is dedicated to the development of numerical methods to capture the important effects due to turbulence. The methods can be grouped into the following three categories:

- **Turbulence models for Reynolds-averaged Navier–Stokes (RANS) equations**: attention is focused on the mean flow and the effects of turbulence on mean flow properties. Prior to the application of numerical methods the Navier–Stokes equations are time averaged (or ensemble averaged in flows with time-dependent boundary conditions). Extra terms appear in the time-averaged (or Reynolds-averaged) flow equations due to the interactions between various turbulent fluctuations. These extra terms are modelled with classical turbulence models: among the best known ones are the k–ε model and

the Reynolds stress model. The computing resources required for reasonably accurate flow computations are modest, so this approach has been the mainstay of engineering flow calculations over the last three decades.

- **Large eddy simulation**: this is an intermediate form of turbulence calculations which tracks the behaviour of the larger eddies. The method involves space filtering of the unsteady Navier–Stokes equations prior to the computations, which passes the larger eddies and rejects the smaller eddies. The effects on the resolved flow (mean flow plus large eddies) due to the smallest, unresolved eddies are included by means of a so-called sub-grid scale model. Unsteady flow equations must be solved, so the demands on computing resources in terms of storage and volume of calculations are large, but (at the time of writing) this technique is starting to address CFD problems with complex geometry.
- **Direct numerical simulation (DNS)**: these simulations compute the mean flow and all turbulent velocity fluctuations. The unsteady Navier–Stokes equations are solved on spatial grids that are sufficiently fine that they can resolve the Kolmogorov length scales at which energy dissipation takes place and with time steps sufficiently small to resolve the period of the fastest fluctuations. These calculations are highly costly in terms of computing resources, so the method is not used for industrial flow computations.

In the next section we discuss the main features and achievements of each of these methods.

3.7 Reynolds-averaged Navier–Stokes equations and classical turbulence models

For most engineering purposes it is unnecessary to resolve the details of the turbulent fluctuations. CFD users are almost always satisfied with information about the time-averaged properties of the flow (e.g. mean velocities, mean pressures, mean stresses etc.). Therefore, the vast majority of turbulent flow computations has been and for the foreseeable future will continue to be carried out with procedures based on the **Reynolds-averaged Navier–Stokes (RANS)** equations (3.30), (3.31a–c) and (3.32). A description of the **effects of turbulence** on the mean flow is nevertheless needed because the time-averaging operation on the momentum equations discards all details concerning the state of the flow contained in the instantaneous fluctuations. We have already seen in section 3.5 that this yields six additional unknowns in the time-averaged momentum equations (3.31a–c): the Reynolds stresses $-\rho\overline{u'^2}$, $-\rho\overline{v'^2}$, $-\rho\overline{w'^2}$, $-\rho\overline{u'v'}$, $-\rho\overline{u'w'}$, $-\rho\overline{v'w'}$. Similarly, time-average scalar transport equations show extra terms containing $\overline{u'\varphi'}$, $\overline{v'\varphi'}$ and $\overline{w'\varphi'}$.

In order to be able to compute turbulent flows with the RANS equations it is necessary to develop **turbulence models** to predict the Reynolds stresses and the scalar transport terms and close the system of mean flow equations (3.30), (3.31a–c) and (3.32). For a turbulence model to be useful in a general-purpose CFD code it must have wide applicability, be accurate, simple and economical to run. The most common RANS turbulence models are classified on the basis of the number of additional transport equations that need to be solved along with the RANS flow equations:

No. of extra transport equations	*Name*
Zero	Mixing length model
One	Spalart–Allmaras model
Two	k–ε model
	k–ω model
	Algebraic stress model
Seven	Reynolds stress model

These models form the basis of standard turbulence calculation procedures in currently available commercial CFD codes.

Eddy viscosity and eddy diffusivity

Of the tabulated models the mixing length and k–ε models are at present by far the most widely used and validated. They are based on the presumption that there exists an analogy between the action of viscous stresses and Reynolds stresses on the mean flow. Both stresses appear on the right hand side of the momentum equation, and in Newton's law of viscosity the viscous stresses are taken to be proportional to the rate of deformation of fluid elements. For an incompressible fluid this gives

$$\tau_{ij} = \mu s_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{2.31}$$

In order to simplify the notation the so-called suffix notation has been used here. The convention of this notation is that i or $j = 1$ corresponds to the x-direction, i or $j = 2$ the y-direction and i or $j = 3$ the z-direction. So, for example,

$$\tau_{12} = \tau_{xy} = \mu\left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1}\right) = \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)$$

In section 3.4 we reviewed experimental evidence which showed that turbulence decays unless there is shear in isothermal incompressible flows. Furthermore, turbulent stresses are found to increase as the mean rate of deformation increases. **Boussinesq** proposed in 1877 that Reynolds stresses might be proportional to mean rates of deformation. Using the suffix notation we get

$$\boxed{\tau_{ij} = -\rho\overline{u_i'u_j'} = \mu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\rho k\delta_{ij}} \tag{3.33}$$

where $k = \frac{1}{2}(\overline{u'^2} + \overline{v'^2} + \overline{w'^2})$ is the turbulent kinetic energy per unit mass (see section 3.3).

The first term of the right hand side is analogous to formula (2.31) above except for the appearance of the turbulent or eddy viscosity μ_t (dimensions Pa s). There is also a kinematic turbulent or eddy viscosity denoted by $\nu_t = \mu_t/\rho$, with dimensions m^2/s. The second term on the right hand side involves δ_{ij}, the Kronecker delta ($\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$). This contribution ensures that the formula gives the correct result for the normal Reynolds

stresses (those with $i = j$), and hence for $\tau_{xx} = -\rho\overline{u'^2}$, $\tau_{yy} = -\rho\overline{v'^2}$ and $\tau_{zz} = -\rho\overline{w'^2}$. To demonstrate the necessity of the extra term we consider an incompressible flow and explore the behaviour of the first part of (3.33) by itself. If we sum this over all the normal stresses (i.e. let $i = 1$, 2 and 3 whilst keeping $i = j$) we find, using continuity, that it is zero, since

$$2\mu_t S_{ii} = 2\mu_t\left[\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z}\right] = 2\mu_t \text{ div } \mathbf{U} = 0$$

Clearly in any flow the sum of the normal stresses $-\rho(\overline{u'^2} + \overline{v'^2} + \overline{w'^2})$ is equal to minus twice the turbulence kinetic energy per unit volume ($-2\rho k$). In equation (3.33) an equal third is allocated to each normal stress component to ensure their sum always has its physically correct value. It should be noted that this implies an isotropic assumption for the normal Reynolds stresses which the data in section 3.4 have shown is inaccurate even in simple two-dimensional flows.

Turbulent transport of heat, mass and other scalar properties can be modelled similarly. Formula (3.33) shows that turbulent momentum transport is assumed to be proportional to mean gradients of velocity (i.e. gradients of momentum per unit mass). By analogy turbulent transport of a scalar is taken to be proportional to the gradient of the mean value of the transported quantity. In suffix notation we get

$$-\rho\overline{u_i'\varphi'} = \Gamma_t \frac{\partial \Phi}{\partial x_i} \tag{3.34}$$

where Γ_t is the turbulent or eddy diffusivity.

Since turbulent transport of momentum and heat or mass is due to the same mechanism – eddy mixing – we expect that the value of the turbulent diffusivity Γ_t is fairly close to that of the turbulent viscosity μ_t. This assumption is better known as the Reynolds analogy. We introduce a turbulent Prandtl/Schmidt number defined as follows:

$$\sigma_t = \frac{\mu_t}{\Gamma_t} \tag{3.35}$$

Experiments in many flows have established that this ratio is often nearly constant. Most CFD procedures assume this to be the case and use values of σ_t around unity.

Preamble

It has become clear from our discussions of simple turbulent flows in section 3.4 that turbulence levels and turbulent stresses vary from point to point in a flow. **Mixing length models** attempt to describe the stresses by means of simple algebraic formulae for μ_t as a function of position. The ***k*–*ε* model** is a more sophisticated and general, but also more costly, description of turbulence which allows for the effects of transport of turbulence properties by convection and diffusion and for production and destruction of turbulence. Two transport equations (PDEs), one for the turbulent kinetic energy k and a further one for the rate of dissipation of turbulent kinetic energy ε, are solved.

The underlying assumption of both these models is that the turbulent viscosity μ_t is isotropic: in other words that the ratio between Reynolds stress

and mean rate of deformation is the same in all directions. This assumption fails in many complex flows where it leads to inaccurate predictions. Here it is necessary to derive and solve transport equations for the Reynolds stresses themselves. It may at first seem strange to think that a stress can be subject to transport. However, it is only necessary to remember that the Reynolds stresses initially appeared on the left hand side of the momentum equations and are physically due to convective momentum exchanges as a consequence of turbulent velocity fluctuations. Fluid momentum – mean momentum as well as fluctuating momentum – can be transported by fluid particles and therefore the Reynolds stresses can also be transported.

The six transport equations, one for each Reynolds stress, contain diffusion, pressure–strain and dissipation terms whose individual effects are unknown and cannot be measured. In **Reynolds stress equation models** (also known in the literature as second-order or second-moment closure models) assumptions are made about these unknown terms, and the resulting PDEs are solved in conjunction with the transport equation for the rate of dissipation of turbulent kinetic energy ε. The design of Reynolds stress equation models is an area of vigorous research, and the models have not been validated as widely as the mixing length and k–ε model. Solving the seven extra PDEs gives rise to a substantial increase in the cost of CFD simulations when compared with the k–ε model, so the application of Reynolds stress equation models outside the academic fraternity is relatively recent.

A much more far-reaching set of modelling assumptions reduces the PDEs describing Reynolds stress transport to algebraic equations to be solved alongside the k and ε equations of the k–ε model. This approach leads to the **algebraic stress models** that are the most economical form of Reynolds stress model able to introduce anisotropic turbulence effects into CFD simulations.

In the following sections the mixing length and k–ε models will be discussed in detail and the main features of the Reynolds stress equation and algebraic stress models will be outlined. We also describe the ***k*–*ω* models** and the **Spalart–Allmaras model**, which are more recent entrants to the industrial CFD arena, and outline the distinguishing features of other models that are beginning to make an impact on industrial turbulence modelling.

3.7.1 Mixing length model

On dimensional grounds we assume the kinematic turbulent viscosity ν_t, which has dimensions m^2/s, can be expressed as a product of a turbulent velocity scale ϑ (m/s) and a turbulent length scale ℓ (m). If *one velocity scale and one length scale* suffice to describe the effects of turbulence, dimensional analysis yields

$$\nu_t = C\vartheta\ell \tag{3.36}$$

where C is a dimensionless constant of proportionality. Of course the dynamic turbulent viscosity is given by

$$\mu_t = C\rho\vartheta\ell$$

Most of the kinetic energy of turbulence is contained in the largest eddies, and turbulence length scale ℓ is therefore characteristic of these eddies which interact with the mean flow. If we accept that there is a strong connection

between the mean flow and the behaviour of the largest eddies we can attempt to link the characteristic velocity scale of the eddies with the mean flow properties. This has been found to work well in simple two-dimensional turbulent flows where the only significant Reynolds stress is $\tau_{xy} = \tau_{yx} = -\rho\overline{u'v'}$ and the only significant mean velocity gradient is $\partial U/\partial y$. For such flows it is at least dimensionally correct to state that, if the eddy length scale is ℓ,

$$\vartheta = c\,\ell\left|\frac{\partial U}{\partial y}\right| \tag{3.37}$$

where c is a dimensionless constant. The absolute value is taken to ensure that the velocity scale is always a positive quantity irrespective of the sign of the velocity gradient.

Combining (3.36) and (3.37) and absorbing the two constants C and c into a new length scale ℓ_m we obtain

$$\boxed{\nu_t = \ell_m^2\left|\frac{\partial U}{\partial y}\right|} \tag{3.38}$$

This is **Prandtl's mixing length model**. Using formula (3.33) and noting that $\partial U/\partial y$ is the only significant mean velocity gradient, the turbulent Reynolds stress is described by

$$\boxed{\tau_{xy} = \tau_{yx} = -\rho\overline{u'v'} = \rho\ell_m^2\left|\frac{\partial U}{\partial y}\right|\frac{\partial U}{\partial y}} \tag{3.39}$$

Turbulence is a function of the flow, and if the turbulence changes it is necessary to account for this within the mixing length model by varying ℓ_m. For a substantial category of simple turbulent flows, which includes the free turbulent flows and wall boundary layers discussed in section 3.4, the turbulence structure is sufficiently simple that ℓ_m can be described by means of simple algebraic formulae. Some examples are given in Table 3.2 (Rodi, 1980).

Table 3.2 Mixing lengths for two-dimensional turbulent flows

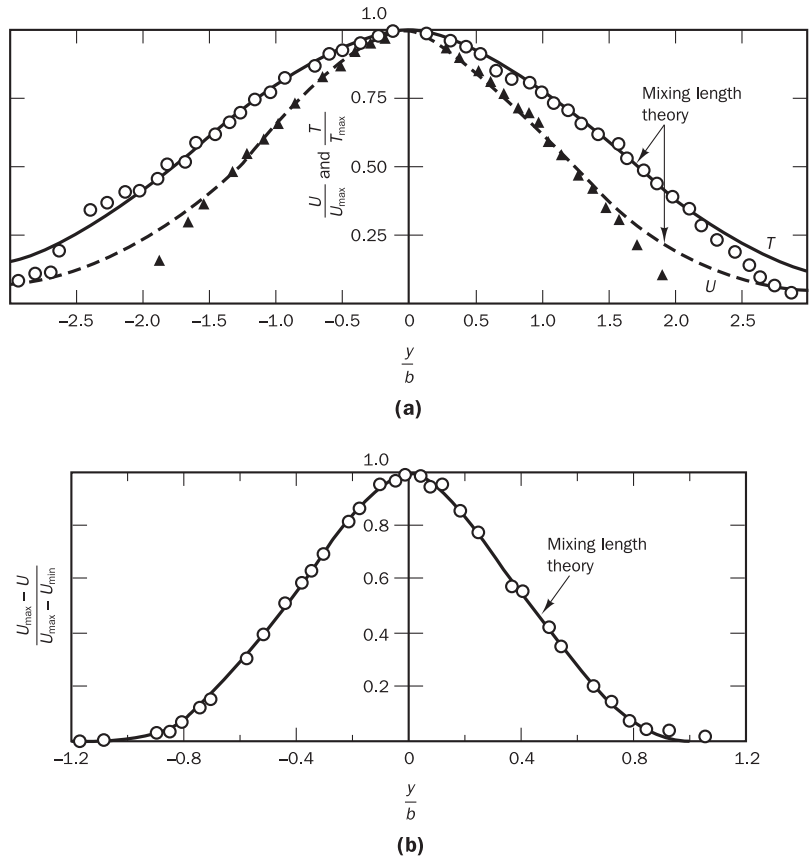
Flow	Mixing length ℓ_m	L
Mixing layer	$0.07L$	Layer width
Jet	$0.09L$	Jet half width
Wake	$0.16L$	Wake half width
Axisymmetric jet	$0.075L$	Jet half width
Boundary layer ($\partial p/\partial x = 0$)		
viscous sub-layer and	$\kappa y[1 - \exp(-y^+/26)]$	
log-law layer ($y/L \le 0.22$)		Boundary layer
outer layer ($y/L \ge 0.22$)	$0.09L$	thickness
Pipes and channels		Pipe radius or
(fully developed flow)	$L[0.14 - 0.08(1 - y/L)^2 - 0.06(1 - y/L)^4]$	channel half width

The mixing length model can also be used to predict turbulent transport of scalar quantities. The only turbulent transport term which matters in the two-dimensional flows for which the mixing length is useful is modelled as follows:

$$-\rho \overline{v'\varphi'} = \Gamma_t \frac{\partial \Phi}{\partial y} \tag{3.40}$$

where $\Gamma_t = \mu_t/\sigma_t$ and $\mu_t = \rho \nu_t$ where ν_t is found from (3.38). Rodi (1980) recommended values for σ_t of 0.9 in near-wall flows, 0.5 for jets and mixing layers and 0.7 in axisymmetric jets.

In the formulae in Table 3.2 y represents the distance from the wall and $\kappa = 0.41$ is von Karman's constant. The expressions give very good agreement between computed results and experiments for mean velocity distributions, wall friction coefficients and other flow properties such as heat transfer coefficients in simple two-dimensional flows. Results for two flows from Schlichting (1979) are given below in Figures 3.14a–b.

Figure 3.14 Results of calculations using mixing length model for (a) planar jet and (b) wake behind a long, slender, circular cylinder *Source*: Schlichting, H. (1979) *Boundary Layer Theory*, 7th edn, reproduced with permission of The McGraw-Hill Companies

The mixing length has been found to be very useful in simple two-dimensional flows with slow changes in the flow direction. In these cases the production of turbulence is in balance with its dissipation throughout

the flow, and turbulence properties develop in proportion with a mean flow length scale L. This means that in such flows the mixing length ℓ_m is proportional to L and can be described as a function of position by means of a simple algebraic formula. The majority of practically important flows, however, involve additional contributions to the budgets of turbulence properties due to transport, i.e. convection and diffusion. Moreover, the production and destruction processes may be modified by the flow itself. Consequently, the mixing length model is not used on its own in general-purpose CFD, but we will find it embedded in many of the more sophisticated turbulence models to describe near-wall flow behaviour as part of the treatment of wall boundary conditions.

An overall assessment of the mixing length model is given in Table 3.3.

Table 3.3 Mixing length model assessment

Advantages:
- easy to implement and cheap in terms of computing resources
- good predictions for thin shear layers: jets, mixing layers, wakes and boundary layers
- well established

Disadvantages:
- completely incapable of describing flows with separation and recirculation
- only calculates mean flow properties and turbulent shear stress

3.7.2 The *k*–*ε* model

In two-dimensional thin shear layers the changes in the flow direction are always so slow that the turbulence can adjust itself to local conditions. In flows where convection and diffusion cause significant differences between production and destruction of turbulence, e.g. in recirculating flows, a compact algebraic prescription for the mixing length is no longer feasible. The way forward is to consider statements regarding the dynamics of turbulence. The k–ε model focuses on the mechanisms that affect the turbulent kinetic energy.

Some preliminary definitions are required first. The instantaneous kinetic energy $k(t)$ of a turbulent flow is the sum of the mean kinetic energy $K = \frac{1}{2}(U^2 + V^2 + W^2)$ and the turbulent kinetic energy $k = \frac{1}{2}(\overline{u'^2} + \overline{v'^2} + \overline{w'^2})$:

$$k(t) = K + k$$

In the developments below we extensively need to use the rate of deformation and the turbulent stresses. To facilitate the subsequent calculations it is common to write the components of the rate of deformation s_{ij} and the stresses τ_{ij} in tensor (matrix) form:

$$s_{ij} = \begin{bmatrix} s_{xx} & s_{xy} & s_{xz} \\ s_{yx} & s_{yy} & s_{yz} \\ s_{zx} & s_{zy} & s_{zz} \end{bmatrix} \quad \text{and} \quad \tau_{ij} = \begin{bmatrix} \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \end{bmatrix}$$

Decomposition of the rate of deformation of a fluid element in a turbulent flow into a mean and a fluctuating component, $s_{ij}(t) = S_{ij} + s'_{ij}$, gives the following matrix elements:

$$s_{xx}(t) = S_{xx} + s'_{xx} = \frac{\partial U}{\partial x} + \frac{\partial u'}{\partial x} \qquad s_{yy}(t) = S_{yy} + s'_{yy} = \frac{\partial V}{\partial y} + \frac{\partial v'}{\partial y}$$

$$s_{zz}(t) = S_{zz} + s'_{zz} = \frac{\partial W}{\partial z} + \frac{\partial w'}{\partial z}$$

$$s_{xy}(t) = S_{xy} + s'_{xy} = s_{yx}(t) = S_{yx} + s'_{yx} = \frac{1}{2}\left[\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right] + \frac{1}{2}\left[\frac{\partial u'}{\partial y} + \frac{\partial v'}{\partial x}\right]$$

$$s_{xz}(t) = S_{xz} + s'_{xz} = s_{zx}(t) = S_{zx} + s'_{zx} = \frac{1}{2}\left[\frac{\partial U}{\partial z} + \frac{\partial W}{\partial x}\right] + \frac{1}{2}\left[\frac{\partial u'}{\partial z} + \frac{\partial w'}{\partial x}\right]$$

$$s_{yz}(t) = S_{yz} + s'_{yz} = s_{zy}(t) = S_{zy} + s'_{zy} = \frac{1}{2}\left[\frac{\partial V}{\partial z} + \frac{\partial W}{\partial y}\right] + \frac{1}{2}\left[\frac{\partial v'}{\partial z} + \frac{\partial w'}{\partial y}\right]$$

The product of a vector **a** and a tensor b_{ij} is a vector **c** whose components can be calculated by application of the ordinary rules of matrix algebra:

$$\mathbf{a}b_{ij} \equiv a_i b_{ij} = [a_1\ a_2\ a_3]\begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} a_1 b_{11} + a_2 b_{21} + a_3 b_{31} \\ a_1 b_{12} + a_2 b_{22} + a_3 b_{32} \\ a_1 b_{13} + a_2 b_{23} + a_3 b_{33} \end{bmatrix}^T = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}^T = c_j = \mathbf{c}$$

The scalar product of two tensors a_{ij} and b_{ij} is evaluated as follows:

$$a_{ij} \,.\, b_{ij} = a_{11}b_{11} + a_{12}b_{12} + a_{13}b_{13} + a_{21}b_{21} + a_{22}b_{22} + a_{23}b_{23} + a_{31}b_{31} + a_{32}b_{32} + a_{33}b_{33}$$

We have used the convention of the suffix notation where the x-direction is denoted by subscript 1, the y-direction by 2 and the z-direction by 3. It can be seen that products are formed by taking the sum over all possible values of every repeated suffix.

Governing equation for mean flow kinetic energy K

An equation for the mean kinetic energy K can be obtained by multiplying x-component Reynolds equation (3.27a) by U, y-component equation (3.27b) by V and z-component equation (3.27c) by W. After adding together the results and a fair amount of algebra it can be shown that the time-average equation governing the mean kinetic energy of the flow is as follows (Tennekes and Lumley, 1972):

$$\underset{\text{(I)}}{\frac{\partial(\rho K)}{\partial t}} + \underset{\text{(II)}}{\text{div}(\rho K\mathbf{U})} = \text{div}(\underset{\text{(III)}}{-P\mathbf{U}} + \underset{\text{(IV)}}{2\mu\mathbf{U}S_{ij}} \underset{\text{(V)}}{- \rho\mathbf{U}\overline{u'_i u'_j}}) \underset{\text{(VI)}}{- 2\mu S_{ij} \,.\, S_{ij}} + \underset{\text{(VII)}}{\rho\overline{u'_i u'_j} \,.\, S_{ij}} \tag{3.41}$$

Or in words

Rate of change of mean kinetic energy K	+	Transport of K by convection	=	Transport of K by pressure	+	Transport of K by viscous stresses	+	Transport of K by Reynolds stress	−	Rate of viscous dissipation of K	−	Rate of destruction of K due to turbulence production

The transport terms (III), (IV) and (V) are all characterised by the appearance of div and it is common practice to place them together inside one pair of

brackets. The effects of the viscous stresses on K have been split into two parts: term (IV), the transport of K due to viscous stresses; and term (VI), the viscous dissipation of mean kinetic energy K. The two terms that contain the Reynolds stresses $-\rho\overline{u_i'u_j'}$ account for turbulence effects: term (V) is the turbulent transport of K by means of Reynolds stresses and (VII) is the net decrease of K due to deformation work by Reynolds stresses giving rise to turbulence production. In high Reynolds number flows the turbulent terms (V) and (VII) are always much larger than their viscous counterparts (IV) and (VI).

Governing equation for turbulent kinetic energy k

Multiplication of each of the instantaneous Navier–Stokes equations (3.24a–c) by the appropriate fluctuating velocity components (i.e. x-component equation multiplied by u' etc.) and addition of all the results, followed by a repeat of this process on the Reynolds equations (3.27a–c), subtraction of the two resulting equations and very substantial rearrangement, yields the equation for turbulent kinetic energy k (Tennekes and Lumley, 1972).

$$\underset{\text{(I)}}{\frac{\partial(\rho k)}{\partial t}} + \underset{\text{(II)}}{\mathrm{div}(\rho k\mathbf{U})} = \mathrm{div}(\underset{\text{(III)}}{-\overline{p'\mathbf{u}'}} + \underset{\text{(IV)}}{2\mu\overline{\mathbf{u}'s_{ij}'}} - \underset{\text{(V)}}{\rho\tfrac{1}{2}\overline{u_i'\,.\,u_i'u_j'}}) - \underset{\text{(VI)}}{2\mu\overline{s_{ij}'\,.\,s_{ij}'}} - \underset{\text{(VII)}}{\rho\overline{u_i'u_j'}\,.\,S_{ij}} \tag{3.42}$$

In words

Rate of change of turbulent kinetic energy k	+	Transport of k by convection	=	Transport of k by pressure	+	Transport of k by viscous stresses	+	Transport of k by Reynolds stress	−	Rate of dissipation of k	+	Rate of production of k

Equations (3.41) and (3.42) look very similar in many respects; however, the appearance of primed quantities on the right hand side of the k-equation shows that changes to the turbulent kinetic energy are mainly governed by turbulent interactions. Terms (VII) in both equations are equal in magnitude, but opposite in sign. In two-dimensional thin shear layers we found (see section 3.4) that the only significant Reynolds stress $-\rho\overline{u'v'}$ is usually positive if the main term of S_{ij} in such a flow, the mean velocity gradient $\partial U/\partial y$, is positive. Hence term (VII) gives a positive contribution in the k-equation and represents a production term. In the K-equation, however, the sign is negative, so there the term destroys mean flow kinetic energy. This expresses mathematically the conversion of mean kinetic energy into turbulent kinetic energy.

The viscous dissipation term (VI),

$$-2\mu\overline{s_{ij}'\,.\,s_{ij}'} = -2\mu(\overline{s_{11}'^2} + \overline{s_{22}'^2} + \overline{s_{33}'^2} + 2\overline{s_{12}'^2} + 2\overline{s_{13}'^2} + 2\overline{s_{23}'^2})$$

gives a negative contribution to (3.42) due to the appearance of the sum of squared fluctuating deformation rates s_{ij}'. The dissipation of turbulent kinetic energy is caused by work done by the smallest eddies against viscous stresses. The rate of dissipation per unit volume (VI) is normally written as the product of the density ρ and the rate of dissipation of turbulent kinetic energy per unit mass ε, so

$$\varepsilon = 2\nu\overline{s_{ij}'\,.\,s_{ij}'} \tag{3.43}$$

The dimensions of ε are m^2/s^3. This quantity is of vital importance in the study of turbulence dynamics. It is the destruction term in the turbulent kinetic energy equation, of a similar order of magnitude as the production term and never negligible. When the Reynolds number is high, the viscous transport term (IV) in (3.42) is always very small compared with the turbulent transport term (V) and the dissipation (VI).

The k–ε model equations

It is possible to develop similar transport equations for all other turbulence quantities including the rate of viscous dissipation ε (see Bradshaw *et al.*, 1981). The exact ε-equation, however, contains many unknown and unmeasurable terms. The **standard *k*–ε model** (Launder and Spalding, 1974) has two model equations, one for k and one for ε, based on our best understanding of the relevant processes causing changes to these variables.

We use k and ε to define velocity scale ϑ and length scale ℓ representative of the large-scale turbulence as follows:

$$\vartheta = k^{1/2} \qquad \ell = \frac{k^{3/2}}{\varepsilon}$$

One might question the validity of using the 'small eddy' variable ε to define the 'large eddy' scale ℓ. We are permitted to do this because at high Reynolds numbers the rate at which large eddies extract energy from the mean flow is broadly matched to the rate of transfer of energy across the energy spectrum to small, dissipating, eddies if the flow does not change too rapidly. If this was not the case the energy at some scales of turbulence could grow or diminish without limit. This does not occur in practice and justifies the use of ε in the definition of ℓ.

Applying dimensional analysis we can specify the eddy viscosity as follows:

$$\boxed{\mu_t = C\rho\vartheta\ell = \rho C_\mu \frac{k^2}{\varepsilon}} \tag{3.44}$$

where C_μ is a dimensionless constant.

The standard k–ε model uses the following transport equations for k and ε:

$$\boxed{\frac{\partial(\rho k)}{\partial t} + \operatorname{div}(\rho k\mathbf{U}) = \operatorname{div}\left[\frac{\mu_t}{\sigma_k}\operatorname{grad} k\right] + 2\mu_t S_{ij} \, . \, S_{ij} - \rho\varepsilon} \tag{3.45}$$

$$\boxed{\frac{\partial(\rho\varepsilon)}{\partial t} + \operatorname{div}(\rho\varepsilon\mathbf{U}) = \operatorname{div}\left[\frac{\mu_t}{\sigma_\varepsilon}\operatorname{grad} \varepsilon\right] + C_{1\varepsilon}\frac{\varepsilon}{k}2\mu_t S_{ij} \, . \, S_{ij} - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k}} \tag{3.46}$$

In words the equations are

Rate of change of k or ε	+	Transport of k or ε by convection	=	Transport of k or ε by diffusion	+	Rate of production of k or ε	−	Rate of destruction of k or ε

The equations contain five adjustable constants: C_μ, σ_k, σ_ε, $C_{1\varepsilon}$ and $C_{2\varepsilon}$. The standard k–ε model employs values for the constants that are arrived at by comprehensive data fitting for a wide range of turbulent flows:

$$C_\mu = 0.09 \quad \sigma_k = 1.00 \quad \sigma_\varepsilon = 1.30 \quad C_{1\varepsilon} = 1.44 \quad C_{2\varepsilon} = 1.92 \tag{3.47}$$

The production term in the model k-equation is derived from the exact production term in (3.42) by substitution of (3.33). A modelled form of the principal transport processes in the k- and ε-equation appears on the right hand side. The turbulent transport terms are represented using the gradient diffusion idea introduced earlier in the context of scalar transport (see equation (3.34)). Prandtl numbers σ_k and σ_ε connect the diffusivities of k and ε to the eddy viscosity μ_t. The pressure term (III) of the exact k-equation cannot be measured directly. Its effect is accounted for in equation (3.45) within the gradient diffusion term.

Production and destruction of turbulent kinetic energy are always closely linked. Dissipation rate ε is large where production of k is large. The model equation (3.46) for ε assumes that its production and destruction terms are proportional to the production and destruction terms of the k-equation (3.45). Adoption of such forms ensures that ε increases rapidly if k increases rapidly and that it decreases sufficiently fast to avoid (non-physical) negative values of turbulent kinetic energy if k decreases. The factor ε/k in the production and destruction terms makes these terms dimensionally correct in the ε-equation. Constants $C_{1\varepsilon}$ and $C_{2\varepsilon}$ allow for the correct proportionality between the terms in the k- and ε-equations.

To compute the Reynolds stresses we use the familiar Boussinesq relationship:

$$-\rho\overline{u_i'u_j'} = \mu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\rho k\delta_{ij} = 2\mu_t S_{ij} - \frac{2}{3}\rho k\delta_{ij} \tag{3.48}$$

Boundary conditions

The model equations for k and ε are elliptic by virtue of the gradient diffusion term. Their behaviour is similar to the other elliptic flow equations, which gives rise to the need for the following boundary conditions:

- inlet: distributions of k and ε must be given
- outlet, symmetry axis: $\partial k/\partial n = 0$ and $\partial\varepsilon/\partial n = 0$
- free stream: k and ε must be given or $\partial k/\partial n = 0$ and $\partial\varepsilon/\partial n = 0$
- solid walls: approach depends on Reynolds number (see below)

In exploratory design calculations the detailed boundary condition information required to operate the model may not be available. Industrial CFD users rarely have measurements of k and ε at their disposal. Progress can be made by entering values of k and ε from the literature (e.g. publications referred to in section 3.4) and subsequently exploring the sensitivity of the results to these inlet distributions. If no information is available at all, rough approximations for the inlet distributions for k and ε in internal flows can be obtained from the turbulence intensity T_i and a characteristic length L of

the equipment (equivalent pipe diameter) by means of the following simple assumed forms:

$$k = \frac{2}{3}(U_{ref}T_i)^2 \quad \varepsilon = C_\mu^{3/4}\frac{k^{3/2}}{\ell} \quad \ell = 0.07L$$

The formulae are closely related to the mixing length formulae given above and the universal distributions near a solid wall given below.

The natural choice of boundary conditions for turbulence-free free stream would seem to be $k = 0$ and $\varepsilon = 0$. Inspection of formula (3.44) shows that this would lead to indeterminate values for the eddy viscosity. In practice, small, but finite, values are commonly used, and once again the sensitivity of the results to these arbitrary assumed values needs to be investigated.

At **high Reynolds number** the standard k–ε model (Launder and Spalding, 1974) avoids the need to integrate the model equations right through to the wall by making use of the universal behaviour of near-wall flows discussed in section 3.4. If y is the co-ordinate direction normal to a solid wall, the mean velocity at a point at y_P with $30 < y_P^+ < 500$ satisfies the log-law (3.19), and measurements of turbulent kinetic energy budgets indicate that the rate of turbulence production equals the rate of dissipation. Using these assumptions and the eddy viscosity formula (3.44) it is possible to develop the following **wall functions**, which relate the local wall shear stress (through u_τ) to the mean velocity, turbulence kinetic energy and rate of dissipation:

$$\boxed{u^+ = \frac{U}{u_\tau} = \frac{1}{\kappa}\ln(Ey_P^+) \quad k = \frac{u_\tau^2}{\sqrt{C_\mu}} \quad \varepsilon = \frac{u_\tau^3}{\kappa y}} \tag{3.49}$$

Von Karman's constant $\kappa = 0.41$ and wall roughness parameter $E = 9.8$ for smooth walls. Schlichting (1979) also gives values of E that are valid for rough walls.

For heat transfer we can use a wall function based on the universal near-wall temperature distribution valid at high Reynolds numbers (Launder and Spalding, 1974)

$$\boxed{T^+ \equiv -\frac{(T - T_w)C_p\rho u_\tau}{q_w} = \sigma_{T,t}\left(u^+ + P\left[\frac{\sigma_{T,l}}{\sigma_{T,t}}\right]\right)} \tag{3.50}$$

with T_P = temperature at near-wall point y_P
T_w = wall temperature
C_p = fluid specific heat at constant pressure
q_w = wall heat flux
$\sigma_{T,t}$ = turbulent Prandtl number
$\sigma_{T,l} = \mu C_p/\Gamma_T$ = (laminar or molecular) Prandtl number
Γ_T = thermal conductivity

Finally P is the pee-function, a correction function dependent on the ratio of laminar to turbulent Prandtl numbers (Launder and Spalding, 1974).

At **low Reynolds numbers** the log-law is not valid, so the above-mentioned boundary conditions cannot be used. Modifications to the k–ε model to enable it to cope with low Reynolds number flows are reviewed in Patel *et al.* (1985). Wall damping needs to be applied to ensure that viscous

and

$$C^*_{1\varepsilon} = C_{1\varepsilon} - \frac{\eta(1 - \eta/\eta_0)}{1 + \beta\eta^3} \quad \eta = \frac{k}{\varepsilon}\sqrt{2S_{ij} . S_{ij}} \quad \eta_0 = 4.377 \quad \beta = 0.012$$

Only the constant β is adjustable; the above value is calculated from near-wall turbulence data. All other constants are explicitly computed as part of the RNG process.

The ε-equation has long been suspected as one of the main sources of accuracy limitations for the standard version of the k–ε model and the RSM in flows that experience large rates of deformation. It is, therefore, interesting to note that the model contains a strain-dependent correction term in the constant $C_{1\varepsilon}$ of the production term in the RNG model ε-equation (it can also be presented as a correction to the sink term).

Yakhot *et al.* (1992) report very good predictions of the flow over a backward-facing step. This performance improvement initially aroused considerable interest and a number of commercial CFD codes have now incorporated the RNG version of the k–ε model. Hanjalič (2004) noted that subsequent experience with the model has not always been positive, because the strain parameter η sensitises the RNG model to the magnitude of the strain. Therefore the effect on the dissipation rate ε is the same irrespective of the sign of the strain. This gives the same effect if a duct is strongly contracting or expanding. Thus, the performance of the RNG k–ε model is better than the standard k–ε model for the expanding duct, but actually worse for a contraction with the same area ratio.

Effects of adverse pressure gradients: turbulence models for aerospace applications

Aerodynamic calculations, such as whole-aircraft simulations, involve very complex geometries and phenomena at different length scales induced by geometry (ranging from flows induced by vortex generators to trailing vortices and fuselage wakes). The bulk of the flow will be effectively inviscid, but the structure of the outer flow is affected by the development of viscous boundary layers and wakes, so local effects at small scale can influence the state of the entire flow field. Specification of a mixing length is not possible in flows of such complexity and, as we have seen previously, the k–ε model does not have an unblemished performance record. Leschziner (in Peyret and Krause, 2000) summarises the problems in this context as follows:

- The k–ε model predicts excessive levels of turbulent shear stress, particularly in the presence of adverse pressure gradients (e.g. in curved shear layers) leading to suppression of separation on curved walls
- Grossly excessive levels of turbulence in stagnation/impingement regions giving rise to excessive heat transfer in reattachment regions

In such complex flows the RSM would be expected to be significantly better, but the computational overhead of this method prevents its routine application for the evaluation of complex external flows. Substantial efforts have been made by the CFD community to develop more economical methods for aerospace applications. We discuss the following recent developments:

- Spalart–Allmaras one-equation model
- Wilcox k–ω model
- Menter shear stress transport (SST) k–ω model

Spalart–Allmaras model

The Spalart–Allmaras model involves one transport equation for kinematic eddy viscosity parameter $\tilde{\nu}$ and a specification of a length scale by means of an algebraic formula, and provides economical computations of boundary layers in external aerodynamics (Spalart and Allmaras, 1992). The (dynamic) eddy viscosity is related to $\tilde{\nu}$ by

$$\mu_t = \rho\tilde{\nu}f_{v1} \tag{3.68}$$

Equation (3.68) contains the wall-damping function $f_{v1} = f_{v1}(\tilde{\nu}/\nu)$, which tends to unity for high Reynolds numbers, so the kinematic eddy viscosity parameter $\tilde{\nu}$ is just equal to the kinematic eddy viscosity ν_t in this case. At the wall the damping function f_{v1} tends to zero.

The Reynolds stresses are computed with

$$\tau_{ij} = -\rho\overline{u_i'u_j'} = 2\mu_t S_{ij} = \rho\tilde{\nu}f_{v1}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) \tag{3.69}$$

The transport equation for $\tilde{\nu}$ is as follows:

$$\underset{\text{(I)}}{\frac{\partial(\rho\tilde{\nu})}{\partial t}} + \underset{\text{(II)}}{\mathrm{div}(\rho\tilde{\nu}\mathbf{U})} = \underset{\text{(III)}}{\frac{1}{\sigma_\nu}\mathrm{div}\left[(\mu + \rho\tilde{\nu})\,\mathrm{grad}(\tilde{\nu}) + C_{b2}\rho\frac{\partial\tilde{\nu}}{\partial x_k}\frac{\partial\tilde{\nu}}{\partial x_k}\right]} + \underset{\text{(V)}}{C_{b1}\rho\tilde{\nu}\tilde{\Omega}} - \underset{\text{(VI)}}{C_{w1}\rho\left(\frac{\tilde{\nu}}{\kappa y}\right)^2 f_w} \tag{3.70}$$

(I) (II) (III) (IV) (V) (VI)

Or in words

Rate of change of viscosity parameter $\tilde{\nu}$	+	Transport of $\tilde{\nu}$ by convection	=	Transport of $\tilde{\nu}$ by turbulent diffusion	+	Rate of production of $\tilde{\nu}$	−	Rate of dissipation of $\tilde{\nu}$

In Equation (3.70) the rate of production of $\tilde{\nu}$ is related to the local mean vorticity as follows:

$$\tilde{\Omega} = \Omega + \frac{\tilde{\nu}}{(\kappa y)^2}f_{v2}$$

where $\Omega = \sqrt{2\Omega_{ij}\Omega_{ij}}$ = mean vorticity

and

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right) = \text{mean vorticity tensor}$$

The functions $f_{v2} = f_{v2}(\tilde{\nu}/\nu)$ and $f_w = f_w(\tilde{\nu}/(\tilde{\Omega}\kappa^2 y^2))$ are further wall-damping functions.

In the k–ε model the length scale is found by combining the two transported quantities k and ε: $\ell = k^{3/2}/\varepsilon$. In a one-equation turbulence model the length scale cannot be computed, but must be specified to determine the rate of dissipation of the transported turbulence quantity. Inspection of the destruction term (VI) of equation (3.70) reveals that κy (with y = distance to the solid wall) has been used as the length scale. The length scale κy also enters in the vorticity parameter $\tilde{\Omega}$ and is just equal to the mixing length used in section 3.4 to develop the log-law for wall boundary layers.

The model constants are as follows:

$\sigma_\nu = 2/3$	$\kappa = 0.4187$	$C_{b1} = 0.1355$	$C_{b2} = 0.622$	$C_{w1} = C_{b1} + \kappa^2 \dfrac{1 + C_{b2}}{\sigma_\nu}$

These model constants and three further ones hidden in the wall functions were tuned for external aerodynamic flows, and the model has been shown to give good performance in boundary layers with adverse pressure gradients, which are important for predicting stalled flows. Its suitability to aerofoil applications means that the Spalart–Allmaras model has also attracted an increasing following among the turbomachinery community. In complex geometries it is difficult to define the length scale, so the model is unsuitable for more general internal flows. Moreover, it lacks sensitivity to transport processes in rapidly changing flows.

Wilcox k–ω model

In the k–ε model the kinematic eddy viscosity ν_t is expressed as the product of a velocity scale $\vartheta = \sqrt{k}$ and a length scale $\ell = k^{3/2}/\varepsilon$. The rate of dissipation of turbulence kinetic energy ε is not the only possible length scale determining variable. In fact, many other two-equation models have been postulated. The most prominent alternative is the k–ω model proposed by Wilcox (1988, 1993a,b, 1994), which uses the turbulence frequency $\omega = \varepsilon/k$ (dimensions s^{-1}) as the second variable. If we use this variable the length scale is $\ell = \sqrt{k}/\omega$. The eddy viscosity is given by

$$\mu_t = \rho k/\omega \tag{3.71}$$

The Reynolds stresses are computed as usual in two-equation models with the Boussinesq expression:

$$\tau_{ij} = -\rho\,\overline{u_i' u_j'} = 2\mu_t S_{ij} - \frac{2}{3}\rho k \delta_{ij} = \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\rho k \delta_{ij} \tag{3.72}$$

The transport equation for k and ω for turbulent flows at high Reynolds is as follows:

$$\frac{\partial(\rho k)}{\partial t} + \operatorname{div}(\rho k \mathbf{U}) = \operatorname{div}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\operatorname{grad}(k)\right] + P_k - \beta^* \rho k \omega \tag{3.73}$$

(I) (II) (III) (IV) (V)

where

$$P_k = \left(2\mu_t S_{ij} \,.\, S_{ij} - \frac{2}{3}\rho k \frac{\partial U_i}{\partial x_j}\delta_{ij}\right)$$

is the rate of production of turbulent kinetic energy and

$$\frac{\partial(\rho\omega)}{\partial t} + \operatorname{div}(\rho\omega\mathbf{U}) = \operatorname{div}\left[\left(\mu + \frac{\mu_t}{\sigma_\omega}\right)\operatorname{grad}(\omega)\right] + \gamma_1\left(2\rho S_{ij} \,.\, S_{ij} - \frac{2}{3}\rho\omega\frac{\partial U_i}{\partial x_j}\delta_{ij}\right) - \beta_1 \rho\omega^2 \tag{3.74}$$

Or in words

Rate of change of k or ω	+	Transport of k or ω by convection	=	Transport of k or ω by turbulent diffusion	+	Rate of production of k or ω	−	Rate of dissipation of k or ω

The model constants are as follows:

$\sigma_k = 2.0$	$\sigma_\omega = 2.0$	$\gamma_1 = 0.553$	$\beta_1 = 0.075$	$\beta^* = 0.09$

The k–ω model initially attracted attention because integration to the wall does not require wall-damping functions in low Reynolds number applications. The value of turbulence kinetic energy k at the wall is set to zero. The frequency ω tends to infinity at the wall, but we can specify a very large value at the wall or, following Wilcox (1988), apply a hyperbolic variation $\omega_P = 6\nu/(\beta_1 y_P^2)$ at the near-wall grid point. Practical experience with the model has shown that the results do not depend too much on the precise details of this treatment.

At inlet boundaries the values of k and ω must be specified, and at outlet boundaries the usual zero gradient conditions are used. The boundary condition of ω in a free stream, where turbulence kinetic energy $k \to 0$ and turbulence frequency $\omega \to 0$, is the most problematic one. Equation (3.71) shows that the eddy viscosity μ_t is indeterminate or infinite as $\omega \to 0$, so a small non-zero value of ω must be specified. Unfortunately, results of the model tend to be dependent on the assumed free stream value of ω (Menter, 1992a), which is a serious problem in external aerodynamics and aerospace applications where free stream boundary conditions are used as a matter of routine.

Menter SST k–ω model

Menter (1992a) noted that the results of the k–ε model are much less sensitive to the (arbitrary) assumed values in the free stream, but its near-wall performance is unsatisfactory for boundary layers with adverse pressure gradients. This led him to suggest a hybrid model using (i) a transformation of the k–ε model into a k–ω model in the near-wall region and (ii) the standard k–ε model in the fully turbulent region far from the wall (Menter, 1992a,b, 1994, 1997). The Reynolds stress computation and the k-equation are the same as in Wilcox's original k–ω model, but the ε-equation is transformed into an ω-equation by substituting $\varepsilon = k\omega$. This yields

$$\underset{\text{(I)}}{\frac{\partial(\rho\omega)}{\partial t}} + \underset{\text{(II)}}{\operatorname{div}(\rho\omega\mathbf{U})} = \underset{\text{(III)}}{\operatorname{div}\left[\left(\mu + \frac{\mu_t}{\sigma_{\omega,1}}\right)\operatorname{grad}(\omega)\right]} + \underset{\text{(IV)}}{\gamma_2\left(2\rho S_{ij}\cdot S_{ij} - \frac{2}{3}\rho\omega\frac{\partial U_i}{\partial x_j}\delta_{ij}\right)} - \underset{\text{(V)}}{\beta_2\rho\omega^2} + \underset{\text{(VI)}}{2\frac{\rho}{\sigma_{\omega,2}\,\omega}\frac{\partial k}{\partial x_k}\frac{\partial \omega}{\partial x_k}} \quad (3.75)$$

Comparison with equation (3.74) shows that (3.75) has an extra source term (VI) on the right hand side: the cross-diffusion term, which arises during the $\varepsilon = k\omega$ transformation of the diffusion term in the ε-equation.

Menter *et al.* (2003) summarise a series of modifications to optimise the performance of the SST k–ω model based on experience with the model in general-purpose computation. The main improvements are:

- *Revised model constants*:

$\sigma_k = 1.0$	$\sigma_{\omega,1} = 2.0$	$\sigma_{\omega,2} = 1.17$	$\gamma_2 = 0.44$	$\beta_2 = 0.083$	$\beta^* = 0.09$

- *Blending functions*: Numerical instabilities may be caused by differences in the computed values of the eddy viscosity with the standard k–ε model in the far field and the transformed k–ε model near the wall. Blending functions are used to achieve a smooth transition between the two models. Blending functions are introduced in the equation to modify the cross-diffusion term and are also used for model constants that take value C_1 for the original k–ω model and value C_2 in Menter's transformed k–ε model:

$$C = F_C C_1 + (1 - F_C)C_2 \tag{3.76}$$

Typically, a blending function $F_C = F_C(\ell_t/y, Re_y)$ is a function of the ratio of turbulence $\ell_t = \sqrt{k}/\omega$ and distance y to the wall and of a turbulence Reynolds number $Re_y = y^2\omega/\nu$. The functional form of F_C is chosen so that it (i) is zero at the wall, (ii) tends to unity in the far field and (iii) produces a smooth transition around a distance half way between the wall and the edge of the boundary layer. This way the method now combines the good near-wall behaviour of the k–ω model with the robustness of the k–ε model in the far field in a numerically stable way.

- *Limiters*: The eddy viscosity is limited to give improved performance in flows with adverse pressure gradients and wake regions, and the turbulent kinetic energy production is limited to prevent the build-up of turbulence in stagnation regions. The limiters are as follows:

$$\mu_t = \frac{a_1 \rho k}{\max(a_1\omega, SF_2)} \tag{3.77a}$$

where $S = \sqrt{2S_{ij}S_{ij}}$, a_1 = constant and F_2 is a blending function, and

$$P_k = \min\left(10\beta^*\rho k\omega,\ 2\mu_t S_{ij} \cdot S_{ij} - \frac{2}{3}\rho k \frac{\partial U_i}{\partial x_j}\delta_{ij}\right) \tag{3.77b}$$

Assessment of performance of turbulence models for aerospace applications

- *External aerodynamics*: The Spalart–Allmaras, k–ω and SST k–ω models are all suitable. The SST k–ω model is most general, and tests suggest that it gives superior performance for zero pressure gradient and adverse pressure gradient boundary layers, free shear layers and a NACA4412 aerofoil (Menter, 1992b). However, the original k–ω model was best for the flow over a backward-facing step.
- *General-purpose CFD*: The Spalart–Allmaras model is unsuitable, but the k–ω and SST k–ω models can both be applied. They both have a similar range of strengths and weaknesses as the k–ε model and fail to include accounts of more subtle interactions between turbulent stresses and mean flow when compared with the RSM.

Anisotropy

Two-equation turbulence models (i.e. k–ε, k–ω and other similar models) are incapable of capturing the more subtle relationships between turbulent energy production and turbulent stresses caused by anisotropy of the normal stresses. They also fail to represent correctly the effects on turbulence of extra strains and body forces. The RSM incorporates these effects exactly, but several unknown turbulence processes (pressure–strain correlations, turbulent diffusion of Reynolds stresses, dissipation) need to be modelled, and the computer storage requirements and run times are significantly increased compared with two-equation models. In order to avoid the performance penalty associated with the solution of extra transport equations in the RSM, several attempts have been made to 'sensitise' two-equation models to the more complex effects. The first method to incorporate sensitivity to normal stress anisotropy was the algebraic stress model. Subsequently, the research groups at NASA Langley Research Center (Speziale) in the USA and at UMIST (Launder) in the UK have developed a number of non-linear two-equation models. These models are discussed below.

Algebraic stress equation model

The algebraic stress model (ASM) represents the earliest attempt to find an economical way of accounting for the anisotropy of Reynolds stresses without going to the full length of solving their transport equations. The large computational cost of solving the RSM is caused by the fact that gradients of the Reynolds stresses R_{ij} etc. appear in the convective C_{ij} and diffusive transport terms D_{ij} of Reynolds stress transport equation (3.55). Rodi and colleagues proposed the idea that, if these transport terms are removed or modelled, the Reynolds stress equations reduce to a set of algebraic equations.

The simplest method is to neglect the convection and diffusion terms altogether. In some cases this appears to be sufficiently accurate (Naot and Rodi, 1982; Demuren and Rodi, 1984). A more generally applicable method is to assume that the sum of the convection and diffusion terms of the Reynolds stresses is proportional to the sum of the convection and diffusion terms of turbulent kinetic energy. Hence

$$\frac{D\overline{u_i'u_j'}}{Dt} - D_{ij} \approx \frac{\overline{u_i'u_j'}}{k}\left(\frac{Dk}{Dt} - [\text{transport of } k \text{ (i.e. div) terms}]\right)$$

$$= \frac{\overline{u_i'u_j'}}{k}(-\overline{u_i'u_j'}\ .\ S_{ij} - \varepsilon) \tag{3.78}$$

The terms in the brackets on the right hand side comprise the sum of the rate of production and the rate of dissipation of turbulent kinetic energy from the exact k-equation (3.42). The Reynolds stresses and the turbulent kinetic energy are both turbulence properties and are closely related, so (3.78) is likely not to be too bad an approximation provided that the ratio $\overline{u_i'u_j'}/k$ does not vary too rapidly across the flow. Further refinements may be obtained by relating the transport by convection and diffusion independently to the transport of turbulent kinetic energy.

Introducing approximation (3.78) into the Reynolds stress transport equation (3.55) with production term P_{ij} (3.57), modelled dissipation rate

term (3.60) and pressure–strain interaction term (3.61) on the right hand side yields after some rearrangement the **algebraic stress model**:

$$\boxed{R_{ij} = \overline{u_i' u_j'} = \frac{2}{3}k\delta_{ij} + \alpha_{\text{ASM}}\left(P_{ij} - \frac{2}{3}P\delta_{ij}\right)\frac{k}{\varepsilon}} \tag{3.79}$$

where $\alpha_{\text{ASM}} = \alpha_{\text{ASM}}(P/\varepsilon)$
and P = production rate of turbulence kinetic energy

The factor α_{ASM} must account for all the physics 'lost' in the algebraic approximation. As indicated, it is a function of the ratio of the rates of production and dissipation of turbulence kinetic energy, which will be close to unity in slowly changing flows. The value of α_{ASM} is around 0.25 for swirling flows. Turbulent scalar transport can also be described by algebraic models derived from their full transport equations that were alluded to in section 3.7.3. Rodi (1980) gives further information for the interested reader.

The Reynolds stresses appear on both sides of (3.79) – on the right hand side they are contained within P_{ij} – so (3.79) is a set of six simultaneous algebraic equations for the six unknown Reynolds stresses R_{ij} that can be solved by matrix inversion or iterative techniques if k and ε are known. Therefore, the formulae are solved in conjunction with the standard k–ε model equations (3.44)–(3.47).

Demuren and Rodi (1984) reported the computation of the secondary flow in non-circular ducts with a somewhat more sophisticated version of this model that includes wall corrections for the pressure–strain term and modified values of adjustable constants to get a good match with measured data in nearly homogeneous shear flows and channel flows. They achieved realistic predictions of the primary flow distortions and secondary flow in square and rectangular ducts. The latter is caused by anisotropy of the normal Reynolds stresses and can therefore not be represented by simulations of the same situation with the standard k–ε model.

Table 3.6 ASM assessment

Advantages:

- cheap method to account for Reynolds stress anisotropy
- potentially combines the generality of approach of the RSM (good modelling of buoyancy and rotation effects possible) with the economy of the k–ε model
- successfully applied to isothermal and buoyant thin shear layers
- if convection and diffusion terms are negligible the ASM performs as well as the RSM

Disadvantages:

- only slightly more expensive than the k–ε model (two PDEs and a system of algebraic equations)
- not as widely validated as the mixing length and k–ε models
- same disadvantages as RSM apply
- model is severely restricted in flows where the transport assumptions for convective and diffusive effects do not apply – validation is necessary to define performance limits

Assessment of performance

The ASM is an economical method of incorporating the effects of anisotropy into the calculations of Reynolds stresses, but it does not consistently perform better than the standard k–ε model (Table 3.6). Moreover, the ASM can suffer from stability problems that can be attributed to the appearance of singularities in the factor $\alpha_{ASM} = \alpha_{ASM}(P/\varepsilon)$, which becomes indeterminate in turbulence-free flow regions, i.e. when $P \to 0$ and $\varepsilon \to 0$. Recently the ASM has been rather overshadowed by the development of non-linear eddy viscosity k–ε models, which will be discussed in the next section.

Non-linear k–ε models

Early work on non-linear two-equation models built on an analogy between viscoelastic fluids and turbulent flows first noted by Rivlin (1957) and elaborated by Lumley (1970). Speziale (1987) presented a systematic framework for the development of non-linear k–ε models. The idea is to 'sensitise' the Reynolds stresses through the introduction of additional effects in a mathematically and physically correct form.

The standard k–ε model uses the Boussinesq approximation (3.33) and eddy viscosity expression (3.44). Hence:

$$-\rho\overline{u_i'u_j'} = \tau_{ij} = \tau_{ij}(S_{ij}, k, \varepsilon, \rho) \tag{3.80}$$

This relationship implies that the turbulence characteristics depend on local conditions only, i.e. the turbulence adjusts itself instantaneously as it is convected through the flow domain. The viscoelastic analogy holds that the adjustment does not take place immediately. In addition to the above dependence on mean strain rate S_{ij}, turbulence kinetic energy k, rate of dissipation ε and fluid density ρ the Reynolds stress should also be a function of the rate of change of mean strain following a fluid particle. So,

$$-\rho\overline{u_i'u_j'} = \tau_{ij} = \tau_{ij}\left(S_{ij}, \frac{DS_{ij}}{Dt}, k, \varepsilon, \rho\right) \tag{3.81}$$

When we studied the RSM we noted that τ_{ij} is actually a transported quantity, i.e. subject to rates of change, convective and diffusive redistribution and to production and dissipation. Bringing in a dependence on DS_{ij}/Dt can be regarded as a partial account of Reynolds stress transport, which recognises that the state of turbulence lags behind the rapid changes that disturb the balance between turbulence production and dissipation.

A group of researchers at NASA Langley Research Center led by Speziale have elaborated this idea and proposed a non-linear k–ε model. Their approach involves the derivation of asymptotic expansions for the Reynolds stresses which maintain terms that are quadratic in velocity gradients (Speziale, 1987):

$$\boxed{\tau_{ij} = -\rho\overline{u_iu_j} = -\frac{2}{3}\rho k\delta_{ij} + \rho C_\mu\frac{k^2}{\varepsilon}2S_{ij} - 4C_DC_\mu^2\frac{k^3}{\varepsilon^2}\left(S_{im}\,.\,S_{mj} - \frac{1}{3}S_{mn}\,.\,S_{mn}\delta_{ij} + \overset{o}{S}_{ij} - \frac{1}{3}\overset{o}{S}_{mm}\delta_{ij}\right)} \tag{3.82}$$

where $\overset{o}{S}_{ij} = \dfrac{\partial S_{ij}}{\partial t} + \mathbf{U}\,.\,\mathrm{grad}(S_{ij}) - \left(\dfrac{\partial U_i}{\partial x_m}\,.\,S_{mj} + \dfrac{\partial U_j}{\partial x_m}\,.\,S_{mi}\right)$ and $C_D = 1.68$

The value of adjustable constant C_D was found by calibration with experimental data.

Equation (3.82) is the non-linear extension of the k–ε model to flows with moderate and large strains. Expression (3.48) for the Reynolds stresses in the standard k–ε model can be regarded as a special case of (3.82) at low rates of deformation when terms that are quadratic in velocity gradients may be dropped. Horiuti (1990) argued in favour of a variant of this approach which retains terms up to third-order in velocity gradients.

The precise form of the model arose from the application of a number of powerful constraints on the mathematical shape of the resulting models, most of which were first compiled and formulated by Lumley (1978):

- *Frame invariance*: turbulence models must be expressed in a mathematical form that is independent of the co-ordinate system used for CFD computations and must give a consistent account of interactions between turbulence and time-dependent translations or rotations of the frame of reference
- *Realisability*: the values of turbulence quantities such as $\overline{u_i'^2}$, k and ε cannot be negative and must be constrained to be always greater than zero.

As an aside it should be noted that application of the realisability constraint without the viscoelastic analogy has given rise to the realisable k–ε model with variable $C_\mu = C_\mu(Sk/\varepsilon)$ where $S = \sqrt{2S_{ij}S_{ij}}$ and modification of the ε-equation (see e.g. Shih *et al.*, 1995).

Launder and colleagues at UMIST worked on non-linear k–ε models with the aim of 'sensitising' the model to the anisotropy of normal Reynolds stresses in a way that preserves the spirit of RSM to the extent that this is possible in a two-equation model. Pope (1975) introduced a generalisation of the eddy viscosity hypothesis based on a power series of tensor products of the mean rate of strain $S_{ij} = \frac{1}{2}(\partial U_i/\partial x_j + \partial U_j/\partial x_i)$ and the mean vorticity $\Omega_{ij} = \frac{1}{2}(\partial U_i/\partial x_j - \partial U_j/\partial x_i)$.

The simplest non-linear eddy viscosity model relates the Reynolds stresses to quadratic tensor products of S_{ij} and Ω_{ij}:

$$\begin{aligned}\tau_{ij} = -\rho\overline{u_i'u_j'} &= 2\mu_t S_{ij} - \frac{2}{3}\rho k\delta_{ij} \\ &-C_1\mu_t\frac{k}{\varepsilon}\left(S_{ik}\,.\,S_{jk} - \frac{1}{3}S_{kl}\,.\,S_{kl}\delta_{ij}\right) \\ &-C_2\mu_t\frac{k}{\varepsilon}(S_{ik}\,.\,\Omega_{jk} + S_{jk}\,.\,\Omega_{ik}) \qquad \text{quadratic terms} \\ &-C_3\mu_t\frac{k}{\varepsilon}\left(\Omega_{ik}\,.\,\Omega_{jk} - \frac{1}{3}\Omega_{kl}\,.\,\Omega_{kl}\delta_{ij}\right)\end{aligned} \tag{3.83}$$

The addition of the last three quadratic terms allows the normal Reynolds stresses to be different, so the model has the potential to capture anisotropy effects. The predictive ability of the model is optimised by adjustment of the three additional model constants C_1, C_2 and C_3 along with the five constants of the original k–ε model. Craft *et al.* (1996) demonstrated that it is necessary to introduce cubic tensor products to obtain the correct sensitising

effect for interactions between Reynolds stress production and streamline curvature. They also included:

- Variable C_μ with a functional dependence on local strain rate S_{ij} and vorticity Ω_{ij}
- Ad hoc modification of the ε-equation to reduce the overprediction of the length scale, leading to poor shear stress predictions in separated flows
- Wall-damping functions to enable integration of the k- and ε-equation to the wall through the viscous sub-layer

Leschziner (in Peyret and Krause, 2000) compared the performance of linear and cubic k–ε models with the RSM to demonstrates the performance enhancement for an aerofoil computation at an incidence angle where trailing edge separation has just occurred. The linear k–ε model fails to indicate the stall condition and gives poor accuracy for a range of other boundary layer parameters, whereas the results of the cubic k–ε model are very close to those of the RSM.

3.7.5 Closing remarks – RANS turbulence models

The field of turbulence modelling provides an area of intense research activity for the CFD and fluid engineering communities. In the previous sections we have outlined the modelling strategy of the most prominent RANS turbulence models that are applied in or under development for commercially available general-purpose codes. Behind much of the research effort in advanced turbulence modelling lies the belief that, irrespective of boundary conditions and geometry, there exists a (limited) number of universal features of turbulence, which, when identified correctly, can form the basis of a complete description of flow variables of interest to an engineer. The emphasis must be on the word 'belief', because the very existence of a classical model – based on time-averaged equations – of this kind is contested by a number of renowned experts in the field. Encouraged by, for example, the early successes of the mixing length model in the external aerodynamics field, they favour the development of dedicated models for limited classes of flow. These two viewpoints naturally lead to two distinct lines of research work:

1. Development and optimisation of turbulence models for limited categories of flows
2. The search for a comprehensive and completely general-purpose turbulence model

Industry has many pressing flow problems to solve that will not wait for the conception of a universal turbulence model. The k–ε model is still widely used in industrial applications and produces useful results in spite of earlier observations relating to its limited validity. Fortunately many sectors of industry are specifically interested in a limited class of flows only, e.g. pipe flows for the oil transportation sector, turbines and combustors for power engineering. The large majority of turbulence research consists of case-by-case examination and validation of existing turbulence models for such specific problems.

stresses take over from turbulent Reynolds stresses at low Reynolds numbers and in the viscous sub-layer adjacent to solid walls. The equations of the low Reynolds number k–ε model, which replace (3.44)–(3.46), are given below:

$$\mu_t = \rho C_\mu f_\mu \frac{k^2}{\varepsilon} \tag{3.51}$$

$$\frac{\partial(\rho k)}{\partial t} + \operatorname{div}(\rho k \mathbf{U}) = \operatorname{div}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\operatorname{grad} k\right] + 2\mu_t S_{ij} \cdot S_{ij} - \rho\varepsilon \tag{3.52}$$

$$\frac{\partial(\rho \varepsilon)}{\partial t} + \operatorname{div}(\rho \varepsilon \mathbf{U}) = \operatorname{div}\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\operatorname{grad} \varepsilon\right] + C_{1\varepsilon} f_1 \frac{\varepsilon}{k} 2\mu_t S_{ij} \cdot S_{ij} - C_{2\varepsilon} f_2 \rho \frac{\varepsilon^2}{k} \tag{3.53}$$

The most obvious modification, which is universally made, is to include the molecular viscosity μ in the diffusion terms in (3.52)–(3.53). The constants C_μ, $C_{1\varepsilon}$ and $C_{2\varepsilon}$ in the standard k–ε model are multiplied by wall-damping functions f_μ, f_1 and f_2, respectively, which are themselves functions of the turbulence Reynolds number ($Re_t = \vartheta\ell/\nu = k^2/(\varepsilon\nu)$), $Re_y = k^{1/2}y/\nu$ and/or similar parameters. As an example we quote the Lam and Bremhorst (1981) wall-damping functions:

$$f_\mu = [1 - \exp(-0.0165\ Re_y)]^2 \left(1 + \frac{20.5}{Re_t}\right)$$

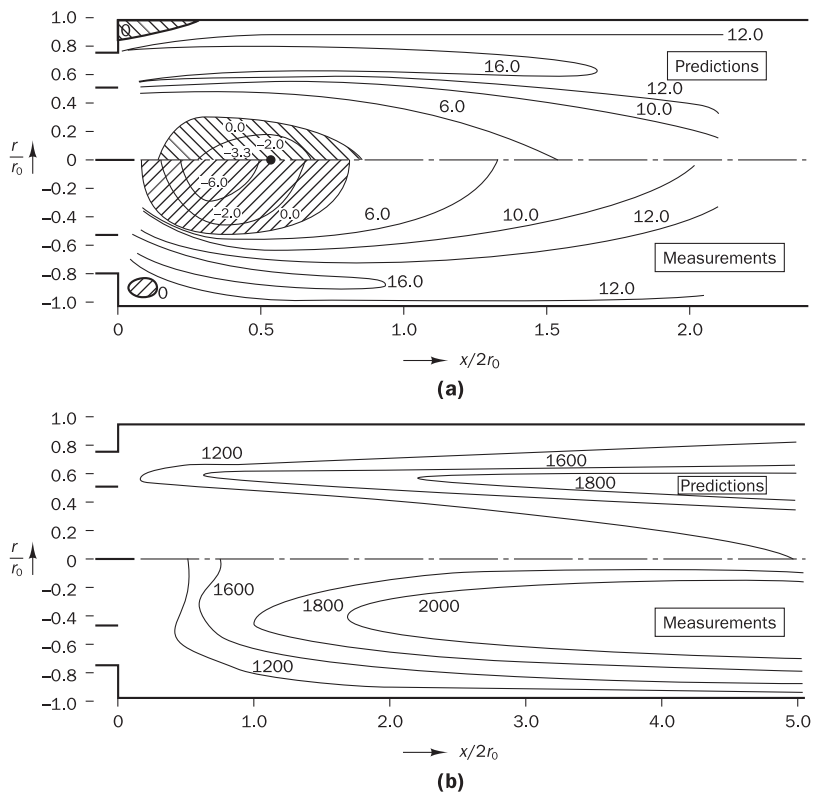
$$f_1 = \left(1 + \frac{0.05}{f_\mu}\right)^3 \qquad f_2 = 1 - \exp(-Re_t^2) \tag{3.54}$$

Equations (3.51)–(3.53) and the RANS equations need to be integrated to the wall, but the boundary condition for ε gives rise to problems. The best available measurements suggest that the rate of dissipation of turbulent energy rises steeply as the wall is approached and tends to an (unknown) constant value. Lam and Bremhorst use $\partial\varepsilon/\partial y = 0$ as the boundary condition. Other low Reynolds number k–ε models are based on a modified dissipation rate variable defined as $\tilde{\varepsilon} = \varepsilon - 2\nu(\partial\sqrt{k}/\partial n)^2$, introduced by Launder and Sharma (1974), which allows us to use the more straightforward boundary condition $\tilde{\varepsilon} = 0$. It should be noted that the resulting equation set is numerically stiff and the further appearance of non-linear wall-damping functions regularly gives rise to severe challenges to achieve convergence.

Assessment of performance

The k–ε model is the most widely used and validated turbulence model. It has achieved notable successes in calculating a wide variety of thin shear layer and recirculating flows without the need for case-by-case adjustment of

the model constants. The model performs particularly well in confined flows where the Reynolds shear stresses are most important. This includes a wide range of flows with industrial engineering applications, which explains its popularity. Versions of the model are available which incorporate effects of buoyancy (Rodi, 1980). Such models are used to study environmental flows such as pollutant dispersion in the atmosphere and in lakes and the modelling of fires. Figure 3.15 (Jones and Whitelaw, 1982) shows the results of early calculations with the k–ε model of turbulent combusting flows for an axisymmetric combustor. Computed contours of axial velocity and temperature are compared with experimental values showing good general agreement but differences in detail. The flow pattern in the combustor is dominated by turbulent transport and hence its correct prediction is vitally important for the development of the flow field and the combustion process. We come back to this issue in Chapter 12 where we examine different models of turbulent combustion.

Figure 3.15 Comparison of predictions of k–ε model with measurements in an axisymmetric combustor: (a) axial velocity contours; (b) temperature contours
Source: Jones and Whitelaw (1982)

In spite of the numerous successes, the standard k–ε model shows only moderate agreement in unconfined flows. The model is reported not to perform well in weak shear layers (far wakes and mixing layers), and the spreading rate of axisymmetric jets in stagnant surroundings is severely overpredicted. In large parts of these flows the rate of production of turbulent kinetic energy is much less than the rate of dissipation, and the difficulties can only be overcome by making ad hoc adjustment to model constants C.

Bradshaw *et al.* (1981) stated that the practice of incorporating the pressure transport term of the exact k-equation in the gradient diffusion expression of

the model equation is deemed to be acceptable on the grounds that the pressure term is sometimes so small that measured turbulent kinetic energy budgets balance without it. They noted, however, that many of these measurements contain substantial errors, and it is certainly not generally true that pressure diffusion effects are negligible.

We can expect that the k–ε model, and all other models that are based on Boussinesq's isotropic eddy viscosity assumption, will have problems in swirling flows and flows with large rapid extra strains (e.g. highly curved boundary layers and diverging passages) that affect the structure of turbulence in a subtle manner. Secondary flows in long non-circular ducts, which are driven by anisotropic normal Reynolds stresses, can also not be predicted due to the same deficiencies of the treatment of normal stresses within the k–ε model. Finally, the model is oblivious to body forces due to rotation of the frame of reference.

A summary of the performance assessment for the standard k–ε model is given in Table 3.4.

Table 3.4 Standard k–ε model assessment

Advantages:
- simplest turbulence model for which only initial and/or boundary conditions need to be supplied
- excellent performance for many industrially relevant flows
- well established, the most widely validated turbulence model

Disadvantages:
- more expensive to implement than mixing length model (two extra PDEs)
- poor performance in a variety of important cases such as:
  - (i) some unconfined flows
  - (ii) flows with large extra strains (e.g. curved boundary layers, swirling flows)
  - (iii) rotating flows
  - (iv) flows driven by anisotropy of normal Reynolds stresses (e.g. fully developed flows in non-circular ducts)

3.7.3 Reynolds stress equation models

The most complex classical turbulence model is the **Reynolds stress equation model (RSM)**, also called the second-order or second-moment closure model. Several major drawbacks of the k–ε model emerge when it is attempted to predict flows with complex strain fields or significant body forces. Under such conditions the individual Reynolds stresses are poorly represented by formula (3.48) even if the turbulent kinetic energy is computed to reasonable accuracy. The exact Reynolds stress transport equation on the other hand can account for the directional effects of the Reynolds stress field.

The modelling strategy originates from work reported in Launder *et al.* (1975). We follow established practice in the literature and call $R_{ij} = -\tau_{ij}/\rho = \overline{u_i' u_j'}$ the Reynolds stress, although the term kinematic Reynolds stress would be more precise. The exact equation for the transport of R_{ij} takes the following form:

$$\frac{DR_{ij}}{Dt} = \frac{\partial R_{ij}}{\partial t} + C_{ij} = P_{ij} + D_{ij} - \varepsilon_{ij} + \Pi_{ij} + \Omega_{ij} \tag{3.55}$$

Rate of change of $R_{ij} = \overline{u_i'u_j'}$	+	Transport of R_{ij} by convection	=	Rate of production of R_{ij}	+	Transport of R_{ij} by diffusion	−	Rate of dissipation of R_{ij}	+	Transport of R_{ij} due to turbulent pressure – strain interactions	+	Transport of R_{ij} due to rotation

Equation (3.55) describes six partial differential equations: one for the transport of each of the six independent Reynolds stresses ($\overline{u_1'^2}$, $\overline{u_2'^2}$, $\overline{u_3'^2}$, $\overline{u_1'u_2'}$, $\overline{u_1'u_3'}$ and $\overline{u_2'u_3'}$, since $\overline{u_2'u_1'} = \overline{u_1'u_2'}$, $\overline{u_3'u_1'} = \overline{u_1'u_3'}$ and $\overline{u_3'u_2'} = \overline{u_2'u_3'}$). If it is compared with the exact transport equation for the turbulent kinetic energy (3.42) two new physical processes appear in the Reynolds stress equations: the pressure–strain interaction or correlation term Π_{ij}, whose effect on the kinetic energy can be shown to be zero, and the rotation term Ω_{ij}.

In CFD computations with the Reynolds stress transport equations the convection, production and rotation terms can be retained in their exact form. The convective term is as follows:

$$C_{ij} = \frac{\partial(\rho U_k \overline{u_i'u_j'})}{\partial x_k} = \text{div}(\rho\, \overline{u_i'u_j'}\, \mathbf{U}) \tag{3.56}$$

the production term is

$$P_{ij} = -\left(R_{im}\frac{\partial U_j}{\partial x_m} + R_{jm}\frac{\partial U_i}{\partial x_m} \right) \tag{3.57}$$

and, finally, the rotational term is given by

$$\Omega_{ij} = -2\omega_k(\overline{u_j'u_m'}\, e_{ikm} + \overline{u_i'u_m'}\, e_{jkm}) \tag{3.58}$$

Here ω_k is the rotation vector and e_{ijk} is the alternating symbol; $e_{ijk} = +1$ if i, j and k are different and in cyclic order, $e_{ijk} = -1$ if i, j and k are different and in anti-cyclic order; and $e_{ijk} = 0$ if any two indices are the same.

To obtain a solvable form of (3.55) we need models for the diffusion, the dissipation rate and the pressure–strain correlation terms on the right hand side. Launder *et al.* (1975) and Rodi (1980) gave comprehensive details of the most general models. For the sake of simplicity we quote those models derived from this approach that are used in some commercial CFD codes. These models often lack somewhat in detail, but their structure is easier to understand and the main message is intact in all cases.

The diffusion term D_{ij} can be modelled with the assumption that the rate of transport of Reynolds stresses by diffusion is proportional to gradients of Reynolds stresses. This gradient diffusion idea recurs throughout turbulence modelling. Commercial CFD codes often favour the simplest form:

$$D_{ij} = \frac{\partial}{\partial x_m}\left(\frac{\nu_t}{\sigma_k}\frac{\partial R_{ij}}{\partial x_m} \right) = \text{div}\left(\frac{\nu_t}{\sigma_k}\ \text{grad}(R_{ij}) \right) \tag{3.59}$$

with $\nu_t = C_\mu \dfrac{k^2}{\varepsilon}$, $C_\mu = 0.09$ and $\sigma_k = 1.0$

The dissipation rate ε_{ij} is modelled by assuming isotropy of the small dissipative eddies. It is set so that it affects the normal Reynolds stresses ($i = j$) only and each stress component in equal measure. This can be achieved by

$$\boxed{\varepsilon_{ij} = \frac{2}{3}\varepsilon\delta_{ij}} \tag{3.60}$$

where ε is the dissipation rate of turbulent kinetic energy defined by (3.43). The Kronecker delta δ_{ij} is given by $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$.

The pressure–strain interactions constitute one of the most important terms in (3.55), but the most difficult one to model accurately. Their effect on the Reynolds stresses is caused by two distinct physical processes: (i) a 'slow' process that reduces anisotropy of the turbulent eddies due to their mutual interactions; and (ii) a 'rapid' process due to interactions between turbulent fluctuations and the mean flow strain that produce the eddies such that the anisotropic production of turbulent eddies is opposed. The overall effect of both processes is to redistribute energy amongst the normal Reynolds stresses ($i = j$) so as to make them more isotropic and to reduce the Reynolds shear stresses ($i \neq j$). The simplest account of the slow process takes the rate of return to isotropic conditions to be proportional to the degree of anisotropy a_{ij} of the Reynolds stresses ($a_{ij} = R_{ij} - \frac{2}{3}k\delta_{ij}$) divided by a characteristic time scale of the turbulence k/ε. The rate of the rapid process is taken to be proportional to the production processes that generate the anisotropy. The simplest representation of the pressure–strain term in the Reynolds stress transport equation is therefore given by

$$\boxed{\Pi_{ij} = -C_1\frac{\varepsilon}{k}\left(R_{ij} - \frac{2}{3}k\delta_{ij}\right) - C_2\left(P_{ij} - \frac{2}{3}P\delta_{ij}\right)} \tag{3.61}$$

with $C_1 = 1.8$ and $C_2 = 0.6$

More advanced accounts include corrections in the second set of brackets in equation (3.61) to ensure that the model is frame invariant (i.e. the effect is the same irrespective of the co-ordinate system).

The effect of the pressure–strain term (3.61) is to decrease anisotropy of Reynolds stresses (i.e. to equalise the normal stresses $\overline{u_1'^2}$, $\overline{u_2'^2}$ and $\overline{u_3'^2}$), but we have seen in section 3.4 that measurements indicate an increase of the anisotropy of normal Reynolds stresses in the vicinity of a solid wall due to damping of fluctuations in the directions normal to the wall. Hence, additional corrections are needed to account for the influence of wall proximity on the pressure–strain terms. These corrections are different in nature from the wall-damping functions encountered in the k–ε model and need to be applied irrespective of the value of the mean flow Reynolds number. It is beyond the scope of this introduction to give all this detail. The reader is directed to a comprehensive model that accounts for all these effects in Launder *et al.* (1975).

Turbulent kinetic energy k is needed in the above formulae and can be found by simple addition of the three normal stresses:

$$k = \tfrac{1}{2}(R_{11} + R_{22} + R_{33}) = \tfrac{1}{2}\left(\overline{u_1'^2} + \overline{u_2'^2} + \overline{u_3'^2}\right)$$

The six equations for Reynolds stress transport are solved along with a model equation for the scalar dissipation rate ε. Again a more exact form is

found in Launder *et al*. (1975), but the equation from the standard k–ε model is used in commercial CFD for the sake of simplicity:

$$\frac{D\varepsilon}{Dt} = \operatorname{div}\left(\frac{\nu_t}{\sigma_\varepsilon}\operatorname{grad}\varepsilon\right) + C_{1\varepsilon}\frac{\varepsilon}{k}2\nu_t S_{ij}\,.\,S_{ij} - C_{2\varepsilon}\frac{\varepsilon^2}{k} \tag{3.62}$$

where $C_{1\varepsilon} = 1.44$ and $C_{2\varepsilon} = 1.92$

Rate of change of ε	+	Transport of ε by convection	=	Transport of ε by diffusion	+	Rate of production of ε	−	Rate of destruction of ε

The usual boundary conditions for elliptic flows are required for the solution of the Reynolds stress transport equations:

- inlet: specified distributions of R_{ij} and ε
- outlet and symmetry: $\partial R_{ij}/\partial n = 0$ and $\partial\varepsilon/\partial n = 0$
- free stream: $R_{ij} = 0$ and $\varepsilon = 0$ are given or $\partial R_{ij}/\partial n = 0$ and $\partial\varepsilon/\partial n = 0$
- solid wall: use wall functions relating R_{ij} to either k or u_τ^2, e.g. $\overline{u_1'^2} = 1.1k$, $\overline{u_2'^2} = 0.25k$, $\overline{u_3'^2} = 0.66k$, $-\overline{u_1'u_2'} = 0.26k$

In the absence of any information, approximate inlet distributions for R_{ij} may be calculated from the turbulence intensity T_i and a characteristic length L of the equipment (e.g. equivalent pipe diameter) by means of the following assumed relationships:

$$k = \frac{2}{3}(U_{ref}T_i)^2 \quad \varepsilon = C_\mu^{3/4}\frac{k^{3/2}}{\ell} \quad \ell = 0.07L$$

$$\overline{u_1'^2} = k \quad \overline{u_2'^2} = \overline{u_3'^2} = \frac{1}{2}k$$

$$\overline{u_i'u_j'} = 0\ (i \neq j)$$

Expressions such as these should not be used without a subsequent test of the sensitivity of results to the assumed inlet boundary conditions.

For computations at high Reynolds numbers wall-function-type boundary conditions can be used, which are very similar to those of the k–ε model and relate the wall shear stress to mean flow quantities. Near-wall Reynolds stress values are computed from formulae such as $R_{ij} = \overline{u_i'u_j'} = c_{ij}k$, where the c_{ij} are obtained from measurements.

Low Reynolds number modifications to the models can be incorporated to add the effects of molecular viscosity to the diffusion terms and to account for anisotropy in the dissipation rate term in the R_{ij}-equations. Wall-damping functions to adjust the constants of the ε-equation and Launder and Sharma's modified dissipation rate variable $\tilde{\varepsilon} \equiv \varepsilon - 2\nu(\partial k^{1/2}/\partial y)^2$ (see also section 3.7.2) give more realistic modelling near solid walls (Launder and Sharma, 1974). So *et al*. (1991) gave a review of the performance of near-wall treatments where details may be found.

Similar models involving three further model PDEs – one for every turbulent scalar flux $\overline{u_i'\varphi'}$ of equation (3.32) – are available for scalar transport.

The interested reader is referred to Rodi (1980) for further material. Commercial CFD codes may use or give as an alternative the simple expedient of solving a single scalar transport equation and using the Reynolds analogy by adding a turbulent diffusion coefficient $\Gamma_t = \mu_t/\sigma_\phi$ to the laminar diffusion coefficient with a specified value of the Prandtl/Schmidt numbers σ_ϕ around 0.7. Little is known about low Reynolds number modifications to the scalar transport equations in near-wall flows.

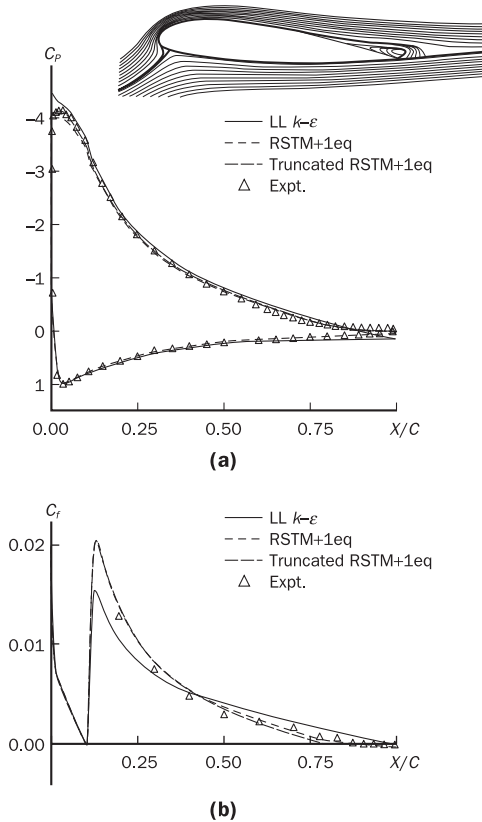
Assessment of performance

RSMs are clearly quite complex, but it is generally accepted that they are the 'simplest' type of model with the potential to describe all the mean flow properties and Reynolds stresses without case-by-case adjustment. The RSM is by no means as well validated as the k–ε model, and because of the high cost of the computations it is not so widely used in industrial flow calculations (Table 3.5). Moreover, the model can suffer from convergence problems due to numerical issues associated with the coupling of the mean velocity and turbulent stress fields through source terms. The extension and improvement of these models is an area of very active research. Once a consensus has been reached about the precise form of the component models and the best numerical solution strategy, it is likely that this form of turbulence modelling will begin to be more widely applied by industrial users. Figure 3.16 (Leschziner, in Peyret and Krause, 2000) gives a performance comparison of the RSM and k–ε models against measured distributions of pressure coefficient and suction-side skin friction coefficients for an Aérospatiale aerofoil. Leschziner notes that the aerofoil is close to stall at the chosen angle of attack. The diagrams show that the k–ε model (labelled LL k–ε) fails to reproduce several details of the pressure distribution in the leading and trailing edge regions. The prediction of the onset of separation depends crucially on the details of the boundary layer structure just upstream, which are captured much better by the RSM model (labelled RSTM + 1eq, to highlight the chosen treatment of the viscous sub-layer). This model also gives excellent agreement with the measured distribution of skin friction on the suction side of the aerofoil.

Table 3.5 RSM assessment

Advantages:
- potentially the most general of all classical turbulence models
- only initial and/or boundary conditions need to be supplied
- very accurate calculation of mean flow properties and *all* Reynolds stresses for many simple and more complex flows including wall jets, asymmetric channel and non-circular duct flows and curved flows

Disadvantages:
- very large computing costs (seven extra PDEs)
- not as widely validated as the mixing length and k–ε models
- performs just as poorly as the k–ε model in some flows due to identical problems with the ε-equation modelling (e.g. axisymmetric jets and unconfined recirculating flows)

Figure 3.16 Comparison of predictions of RSM and standard k–ε model with measurements on a high-lift Aérospatiale aerofoil: (a) pressure coefficient; (b) skin friction coefficient
Source: Leschziner, in Peyret and Krause (2000)

3.7.4 Advanced turbulence models

Two-equation turbulence models, such as the k–ε model introduced earlier, give good results for simple flows and some recirculating flows, but research over a period of three decades has highlighted a number of shortcomings. Leschziner (in Peyret and Krause, 2000) and Hanjalič (2004) summarised the nature and causes of these performance problems:

- *Low Reynolds number flows*: in these flows wall functions based on the log-law are inaccurate and it is necessary to integrate the k- and ε-equations to the wall. Very rapid changes occur in the distributions of k and ε as we reach the buffer layer between the fully turbulent region and the viscous sublayer. This requires large numbers of grid points to resolve the changes, and we also need non-linear wall-damping functions to force upon k and ε the correct behaviour as the character of the near-wall flow changes from turbulence dominated to viscous dominated. As a consequence the system of equations that needs to be solved is numerically stiff, which means that it may be difficult to get converged solutions. Furthermore, the results can be grid dependent.
- *Rapidly changing flows*: the Reynolds stress $-\rho\overline{u_i'u_j'}$ is proportional to the mean rate of strain S_{ij} in two-equation models. This only holds when

the rates of production and dissipation of turbulence kinetic energy are roughly in balance. In rapidly changing flows this is not the case.

- *Stress anisotropy*: the normal Reynolds stresses $-\rho\overline{u_i'^2}$ will all be approximately equal to $-\frac{2}{3}\rho k$ if a thin shear layer flow is evaluated using a two-equation model. Experimental data presented in section 3.4 showed that this is not correct, but in spite of this the k–ε model performs well in such flows because the *gradients* of normal turbulent stresses $-\rho\overline{u_i'^2}$ are small compared with the *gradient* of the dominant turbulent shear stress $-\rho\overline{u'v'}$. Consequently, the normal stresses may be large, but they are not dynamically active in thin shear layer flows, i.e. they are not responsible for driving any flows. In more complex flows the gradients of normal turbulent stresses are not negligible and can drive significant flows. These effects cannot be predicted by the standard two-equation models.
- *Strong adverse pressure gradients and recirculation regions*: this problem particularly affects the k–ε model and is also attributable to the isotropy of its predicted normal Reynolds stresses and the resultant failure to represent correctly the subtle interactions between normal Reynolds stresses and mean flow that determine turbulent energy production. The k–ε model overpredicts the shear stress and suppresses separation in flows over curved walls. This is a significant problem in flows over aerofoils, e.g. in aerospace applications.
- *Extra strains*: streamline curvature, rotation and extra body forces all give rise to additional interactions between the mean strain rate and the Reynolds stresses. These physical effects are not captured by standard two-equation models.

As we have seen, the RSM incorporates an exact representation of the Reynolds stress production process and, hence, addresses most of these problems adequately, but at the cost of a significant increase in computer storage and run time. Below we consider some of the more recent advances in turbulence modelling that seek to address some or all of the above problems.

Advanced treatment of the near-wall region: two-layer k–ε model

The two-layer model represents an improved treatment of the near-wall region for turbulent flows at low Reynolds number. The intention is, as in the low Reynolds number k–ε model discussed earlier, to integrate to the wall by placing the near-wall grid point in the viscous sublayer ($y^+ < 1$). The numerical stability problems (Chen and Patel, 1988) associated with the non-linear wall-damping functions, necessary in the low Reynolds number k–ε model to integrate both k- and ε-equations to the wall, are avoided by sub-dividing the boundary layer into two regions (Rodi, 1991):

- Fully turbulent region, $Re_y = y\sqrt{k}/\nu \geq 200$: the standard k–ε model is used and the eddy viscosity is computed with the usual relationship (3.44), $\mu_{t,t} = C_\mu \rho k^2/\varepsilon$
- Viscous region, $Re_y < 200$: only the k-equation is solved in this region and a length scale is specified using $\ell = \kappa y[1 - \exp(-Re_y/A)]$ for the evaluation of the rate of dissipation with $\varepsilon = C_\mu^{3/4} k^{3/2}/\ell$ using $A = 2\kappa C_\mu^{-3/4}$ and the eddy viscosity in this region with $\mu_{t,v} = C_\mu^{1/4}\rho\sqrt{k}\,\ell$ and $A = 70$

The mixing length formulae are similar in form to the expression in Table 3.2 for the length scale in the viscous sub-layer of a wall boundary layer. In order to avoid instabilities associated with differences between $\mu_{t,t}$ and $\mu_{t,v}$ at the join between the fully turbulent and viscous regions, a blending formula is used to evaluate the eddy viscosity in $\tau_{ij} = -\rho \overline{u_i' u_j'} = 2\mu_t S_{ij} - \frac{2}{3}\rho k \delta_{ij}$:

$$\mu_t = F_\mu \mu_{t,t} + (1 - F_\mu)\mu_{t,v} \tag{3.64}$$

The blending function $F_\mu = F_\mu(Re_y)$ is zero at the wall and tends to 1 in the fully turbulent region when $Re_y \gg 200$. The functional form of F_μ is designed to ensure a smooth transition around $Re_y = 200$.

The two-layer model is less grid dependent and more numerically stable than the earlier low Reynolds number k–ε models and has become quite popular in more complex flow simulations where integration to the wall of the flow equations is necessary.

Strain sensitivity: RNG k–ε model

The statistical mechanics approach has led to new mathematical formalisms, which, in conjunction with a limited number of assumptions regarding the statistics of small-scale turbulence, provide a rigorous basis for the extension of eddy viscosity models. The renormalization group (RNG) devised by Yakhot and Orszag of Princeton University has attracted most interest. They represented the effects of the small-scale turbulence by means of a random forcing function in the Navier–Stokes equation. The RNG procedure systematically removes the small scales of motion from the governing equations by expressing their effects in terms of larger scale motions and a modified viscosity. The mathematics is highly abstruse; we only quote the RNG k–ε model equations for high Reynolds number flows derived by Yakhot *et al.* (1992):

$$\boxed{\frac{\partial(\rho k)}{\partial t} + \mathrm{div}(\rho k \mathbf{U}) = \mathrm{div}[\alpha_k \mu_{eff}\ \mathrm{grad}\ k] + \tau_{ij} . S_{ij} - \rho\varepsilon} \tag{3.65}$$

$$\boxed{\frac{\partial(\rho \varepsilon)}{\partial t} + \mathrm{div}(\rho \varepsilon \mathbf{U}) = \mathrm{div}[\alpha_\varepsilon \mu_{eff}\ \mathrm{grad}\ \varepsilon] + C^*_{1\varepsilon}\frac{\varepsilon}{k}\tau_{ij} . S_{ij} - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k}} \tag{3.66}$$

with

$$\boxed{\tau_{ij} = -\rho\overline{u_i' u_j'} = 2\mu_t S_{ij} - \frac{2}{3}\rho k \delta_{ij}}$$

and

$$\mu_{eff} = \mu + \mu_t \quad \mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$$

and

$$C_\mu = 0.0845 \quad \alpha_k = \alpha_\varepsilon = 1.39 \quad C_{1\varepsilon} = 1.42 \quad C_{2\varepsilon} = 1.68 \tag{3.67}$$

The literature is too extensive even to begin to review here. The main sources of useful, applications-oriented information are: *Transactions of the American Society of Mechanical Engineers* – in particular the *Journal of Fluids Engineering*, *Journal of Heat Transfer* and *Journal of Engineering for Gas Turbines and Power* – as well as the *AIAA Journal*, the *International Journal of Heat and Mass Transfer* and the *International Journal of Heat and Fluid Flow*.

3.8 Large eddy simulation

In spite of century-long efforts to develop RANS turbulence models, a general-purpose model suitable for a wide range of practical applications has so far proved to be elusive. This is to a large extent attributable to differences in the behaviour of large and small eddies. The smaller eddies are nearly isotropic and have a universal behaviour (for turbulent flows at sufficiently high Reynolds numbers at least). On the other hand, the larger eddies, which interact with and extract energy from the mean flow, are more anisotropic and their behaviour is dictated by the geometry of the problem domain, the boundary conditions and body forces. When Reynolds-averaged equations are used the collective behaviour of *all* eddies must be described by a single turbulence model, but the problem dependence of the largest eddies complicates the search for widely applicable models. A different approach to the computation of turbulent flows accepts that the larger eddies need to be computed for each problem with a time-dependent simulation. The universal behaviour of the smaller eddies, on the other hand, should hopefully be easier to capture with a compact model. This is the essence of the **large eddy simulation (LES)** approach to the numerical treatment of turbulence.

Instead of time-averaging, LES uses a spatial filtering operation to separate the larger and smaller eddies. The method starts with the selection of a filtering function and a certain cutoff width with the aim of resolving in an unsteady flow computation all those eddies with a length scale greater than the cutoff width. In the next step the spatial filtering operation is performed on the time-dependent flow equations. During spatial filtering information relating to the smaller, filtered-out turbulent eddies is destroyed. This, and interaction effects between the larger, resolved eddies and the smaller unresolved ones, gives rise to sub-grid-scale stresses or SGS stresses. Their effect on the resolved flow must be described by means of an SGS model. If the finite volume method is used the time-dependent, space-filtered flow equations are solved on a grid of control volumes along with the SGS model of the unresolved stresses. This yields the mean flow and all turbulent eddies at scales larger than the cutoff width. In this section we review the methodology of LES computation of turbulent flows and summarise recent achievements in the calculation of industrially relevant flows.

3.8.1 Spatial filtering of unsteady Navier–Stokes equations

Filters are familiar separation devices in electronics and process applications that are designed to split an input into a desirable, retained part and an undesirable, rejected part. The details of the design of a filter – in particular its functional form and the cutoff width Δ – determine precisely what is retained and rejected.

Filtering functions

In LES we define a spatial filtering operation by means of a **filter function** $G(\mathbf{x}, \mathbf{x}', \Delta)$ as follows:

$$\bar{\phi}(\mathbf{x}, t) \equiv \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(\mathbf{x}, \mathbf{x}', \Delta)\phi(\mathbf{x}', t)\mathrm{d}x_1'\,\mathrm{d}x_2'\,\mathrm{d}x_3' \tag{3.84}$$

where $\bar{\phi}(\mathbf{x}, t)$ = filtered function
and $\phi(\mathbf{x}, t)$ = original (unfiltered) function
and Δ = filter cutoff width

In this section the overbar indicates spatial filtering, not time-averaging. Equation (3.84) shows that filtering is an integration, just like time-averaging in the development of the RANS equations, only in the LES the integration is not carried out in time but in three-dimensional space. It should be noted that filtering is a linear operation.

The commonest forms of the filtering function in three-dimensional LES computations are

- Top-hat or box filter:

$$G(\mathbf{x}, \mathbf{x}', \Delta) = \begin{cases} 1/\Delta^3 & |\mathbf{x} - \mathbf{x}'| \le \Delta / 2 \\ 0 & |\mathbf{x} - \mathbf{x}'| > \Delta / 2 \end{cases} \tag{3.85a}$$

- Gaussian filter:

$$G(\mathbf{x}, \mathbf{x}', \Delta) = \left(\frac{\gamma}{\pi\Delta^2}\right)^{3/2} \exp\left(-\gamma\frac{|\mathbf{x} - \mathbf{x}'|^2}{\Delta^2}\right) \tag{3.85b}$$

typical value for parameter $\gamma = 6$

- Spectral cutoff:

$$G(\mathbf{x}, \mathbf{x}', \Delta) = \prod_{i=1}^{3} \frac{\sin[(x_i - x_i')/\Delta]}{(x_i - x_i')} \tag{3.85c}$$

The top-hat filter is used in finite volume implementations of LES. The Gaussian and spectral cutoff filters are preferred in the research literature. The Gaussian filter was introduced for LES in finite differences by the Stanford group, which, over a period of more than three decades, has been the centre of research on LES and has established a rigorous basis for the technique as a turbulence modelling tool. Spectral methods (i.e. Fourier series to describe the flow variables) are also used in turbulence research, and the spectral filter gives a sharp cutoff in the energy spectrum at a wavelength of Δ/π. The latter is attractive from the point of view of separation of the large and small eddy scales, but the spectral method cannot be used in general-purpose CFD.

The cutoff width is intended as an indicative measure of the size of eddies that are retained in the computations and the eddies that are rejected. In principle, we can choose the **cutoff width** Δ to be any size, but in CFD computations with the finite volume method it is pointless to select a cutoff width that is smaller than the grid size. In this type of computation only a single nodal value of each flow variable is retained on each grid cell, so all finer detail is lost anyway. The most common selection is to take the cutoff

width to be of the same order as the grid size. In three-dimensional computations with grid cells of different length Δx, width Δy and height Δz the cutoff width is often taken to be the cube root of the grid cell volume:

$$\Delta = \sqrt[3]{\Delta x \Delta y \Delta z} \tag{3.86}$$

Filtered unsteady Navier–Stokes equations

As before in section 3.3 we focus our attention on incompressible flows. As usual we take Cartesian co-ordinates so that the velocity vector $\mathbf{u}$ has u-, v-, w-components. The unsteady Navier–Stokes equations for a fluid with constant viscosity μ are as follows:

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) = 0 \tag{2.4}$$

$$\frac{\partial(\rho u)}{\partial t} + \operatorname{div}(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + \mu \operatorname{div}(\operatorname{grad}(u)) + S_u \tag{2.37a}$$

$$\frac{\partial(\rho v)}{\partial t} + \operatorname{div}(\rho v \mathbf{u}) = -\frac{\partial p}{\partial y} + \mu \operatorname{div}(\operatorname{grad}(v)) + S_v \tag{2.37b}$$

$$\frac{\partial(\rho w)}{\partial t} + \operatorname{div}(\rho w \mathbf{u}) = -\frac{\partial p}{\partial z} + \mu \operatorname{div}(\operatorname{grad}(w)) + S_w \tag{2.37c}$$

If the flow is also incompressible we have $\operatorname{div}(\mathbf{u}) = 0$, and hence the viscous momentum source terms S_u, S_v and S_w are zero.

Considerable further simplification of the algebra is possible if we use the same filtering function $G(\mathbf{x}, \mathbf{x}') = G(\mathbf{x} - \mathbf{x}')$ throughout the computational domain, i.e. G is independent of position $\mathbf{x}$. If we use such a uniform filter function we can, by exploiting the linearity of the filtering operation, swap the order of the filtering and differentiation with respect to time, as well as the order of filtering and differentiation with respect to space co-ordinates. We have already seen this commutative property in action in section 3.3 when the time-averaged RANS flow equations were derived. Filtering of equation (2.4) yields the **LES continuity equation**:

$$\boxed{\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \bar{\mathbf{u}}) = 0} \tag{3.87}$$

The overbar in this and all following equations in this section indicates a filtered flow variable.

Repeating the process for equations (2.37a–c) gives

$$\frac{\partial(\rho \bar{u})}{\partial t} + \operatorname{div}(\rho \overline{u\mathbf{u}}) = -\frac{\partial \bar{p}}{\partial x} + \mu \operatorname{div}(\operatorname{grad}(\bar{u})) \tag{3.88a}$$

$$\frac{\partial(\rho \bar{v})}{\partial t} + \operatorname{div}(\rho \overline{v\mathbf{u}}) = -\frac{\partial \bar{p}}{\partial y} + \mu \operatorname{div}(\operatorname{grad}(\bar{v})) \tag{3.88b}$$

$$\frac{\partial(\rho \bar{w})}{\partial t} + \operatorname{div}(\rho \overline{w\mathbf{u}}) = -\frac{\partial \bar{p}}{\partial z} + \mu \operatorname{div}(\operatorname{grad}(\bar{w})) \tag{3.88c}$$

Equation set (3.87) and (3.88a–c) should be solved to yield the filtered velocity field $\bar{u}$, $\bar{v}$, $\bar{w}$ and filtered pressure field $\bar{p}$. We now face the problem that we need to compute convective terms of the form $\text{div}(\rho\overline{\phi\mathbf{u}})$ on the left hand side, but we only have available the filtered velocity field $\bar{u}$, $\bar{v}$, $\bar{w}$ and pressure field $\bar{p}$. To make some progress we write

$$\text{div}(\rho\,\overline{\phi\mathbf{u}}) = \text{div}(\bar{\phi}\bar{\mathbf{u}}) + (\text{div}(\rho\overline{\phi\mathbf{u}}) - \text{div}(\bar{\phi}\bar{\mathbf{u}}))$$

The first term on the right hand side can be calculated from the filtered $\bar{\phi}$ – and $\bar{u}$ – fields and the second term is replaced by a model.

Substitution into (3.88a–c) and some rearrangement yields the **LES momentum equations**:

$$\frac{\partial(\rho\bar{u})}{\partial t} + \text{div}(\rho\bar{u}\bar{\mathbf{u}}) = -\frac{\partial\bar{p}}{\partial x} + \mu\,\text{div}(\text{grad}(\bar{u})) - (\text{div}(\rho\,\overline{u\mathbf{u}}) - \text{div}(\rho\bar{u}\bar{\mathbf{u}})) \tag{3.89a}$$

$$\frac{\partial(\rho\bar{v})}{\partial t} + \text{div}(\rho\bar{v}\bar{\mathbf{u}}) = -\frac{\partial\bar{p}}{\partial y} + \mu\,\text{div}(\text{grad}(\bar{v})) - (\text{div}(\rho\,\overline{v\mathbf{u}}) - \text{div}(\rho\bar{v}\bar{\mathbf{u}})) \tag{3.89b}$$

$$\frac{\partial(\rho\bar{w})}{\partial t} + \text{div}(\rho\bar{w}\bar{\mathbf{u}}) = -\frac{\partial\bar{p}}{\partial z} + \mu\,\text{div}(\text{grad}(\bar{w})) - (\text{div}(\rho\overline{w\mathbf{u}}) - \text{div}(\rho\bar{w}\bar{\mathbf{u}})) \tag{3.89c}$$

(I) (II) (III) (IV) (V)

The filtered momentum equations look very much like the RANS momentum equations (3.26a–c) or (3.27a–c). Terms (I) are the rate of change of the filtered x-, y- and z-momentum. Terms (II) and (IV) are the convective and diffusive fluxes of filtered x-, y- and z-momentum. Terms (III) are the gradients in the x-, y- and z-directions of the filtered pressure field. The last terms (V) are caused by the filtering operation, just like the Reynolds stresses in the RANS momentum equations that arose as a consequence of time-averaging. They can be considered as a divergence of a set of stresses τ_{ij}. In suffix notation the i-component of these terms can be written as follows:

$$\text{div}(\rho\overline{u_i\mathbf{u}} - \rho\bar{u}_i\bar{\mathbf{u}}) = \frac{\partial(\rho\overline{u_iu} - \rho\bar{u}_i\bar{u})}{\partial x} + \frac{\partial(\rho\overline{u_iv} - \rho\bar{u}_i\bar{v})}{\partial y} + \frac{\partial(\rho\overline{u_iw} - \rho\bar{u}_i\bar{w})}{\partial z} = \frac{\partial\tau_{ij}}{\partial x_j} \tag{3.90a}$$

$$\text{where } \tau_{ij} = \rho\overline{u_i\mathbf{u}} - \rho\bar{u}_i\bar{\mathbf{u}} = \rho\,\overline{u_iu_j} - \rho\,\bar{u}_i\,\bar{u}_j \tag{3.90b}$$

In recognition of the fact that a substantial portion of τ_{ij} is attributable to convective momentum transport due to interactions between the unresolved or SGS eddies, these stresses are commonly termed the **sub-grid-scale stresses**. However, unlike the Reynolds stresses in the RANS equations, the LES SGS stresses contain further contributions. The nature of these contributions can be determined with the aid of a decomposition of a flow variable $\phi(\mathbf{x}, t)$ as the sum of (i) the filtered function $\bar{\phi}(\mathbf{x}, t)$ with spatial variations that are larger than the cutoff width and are resolved by the LES computation and (ii) $\phi'(\mathbf{x}, t)$, which contains unresolved spatial variations at a length scale smaller than the filter cutoff width:

$$\phi(\mathbf{x}, t) = \bar{\phi}(\mathbf{x}, t) + \phi'(\mathbf{x}, t) \tag{3.91}$$

Using this decomposition in equation (3.90b) we can write the first term on the far right hand side as follows:

$$\begin{aligned}\rho\overline{u_i u_j} &= \rho\overline{(\bar{u}_i + u_i')(\bar{u}_j + u_j')} = \rho\overline{\bar{u}_i\bar{u}_j} + \rho\overline{\bar{u}_i u_j'} + \rho\overline{u_i'\bar{u}_j} + \rho\overline{u_i' u_j'} \\ &= \rho\bar{u}_i\bar{u}_j + (\rho\overline{\bar{u}_i\bar{u}_j} - \rho\bar{u}_i\bar{u}_j) + \rho\overline{\bar{u}_i u_j'} + \rho\overline{u_i'\bar{u}_j} + \rho\overline{u_i' u_j'}\end{aligned}$$

Now we can write the SGS stresses as follows:

$$\boxed{\tau_{ij} = \rho\overline{u_i u_j} - \rho\bar{u}_i\bar{u}_j = (\rho\overline{\bar{u}_i\bar{u}_j} - \rho\bar{u}_i\bar{u}_j) + \underbrace{\rho\overline{\bar{u}_i u_j'} + \rho\overline{u_i'\bar{u}_j}} + \rho\overline{u_i' u_j'}} \tag{3.92}$$

(I) (II) (III)

Thus, we find that the SGS stresses contain three groups of contributions:

- Term (I), **Leonard stresses** L_{ij}: $L_{ij} = \rho\overline{\bar{u}_i\bar{u}_j} - \rho\bar{u}_i\bar{u}_j$
- Term (II), **cross-stresses** C_{ij}: $C_{ij} = \rho\overline{\bar{u}_i u_j'} + \rho\overline{u_i'\bar{u}_j}$
- Term (III), **LES Reynolds stresses** R_{ij}: $R_{ij} = \rho\overline{u_i' u_j'}$

The Leonard stresses L_{ij} are solely due to effects at resolved scale. They are caused by the fact that a second filtering operation makes a change to a filtered flow variable, i.e. $\bar{\bar{\phi}} \neq \bar{\phi}$ for space-filtered variables, unlike in time-averaging, where $\overline{\overline{\varphi(t)}} = \bar{\Phi} = \Phi = \overline{\varphi(t)}$ (compare equation (3.21)). These stress contributions were named after the American scientist A. Leonard, who first identified an approximate method to compute them from the filtered flow field (see Leonard (1974) for further details). The cross-stresses C_{ij} are due to interactions between the SGS eddies and the resolved flow. An approximate expression for this term is given in Ferziger (1977). Finally, the LES Reynolds stresses R_{ij} are caused by convective momentum transfer due to interactions of SGS eddies and are modelled with a so-called SGS turbulence model. Just like the Reynolds stresses in the RANS equations, the SGS stresses (3.92) must be modelled. Below we discuss the most prominent SGS models.

3.8.2 Smagorinksy–Lilly SGS model

In simple flows such as two-dimensional thin shear layers the Boussinesq eddy viscosity hypothesis (3.33) was often found to give good predictions of Reynolds-averaged turbulent stresses. In recognition of the intimate connection between turbulence production and mean strain, the hypothesis takes the turbulent stresses to be proportional to the mean rate of strain. Success of the approach requires that (i) the changes in the flow direction should be slow so that production and dissipation of turbulence are more or less in balance and (ii) the turbulence structure should be isotropic (or if this is not the case the gradients of the anisotropic normal stresses should not be dynamically active). Smagorinsky (1963) suggested that, since the smallest turbulent eddies are almost isotropic, we expect that the Boussinesq hypothesis might provide a good description of the effects of the unresolved eddies on the resolved flow. Thus, in Smagorinsky's SGS model the **local SGS stresses** R_{ij} are taken to be **proportional to the local rate of strain of the resolved flow** $\bar{S}_{ij} = \frac{1}{2}(\partial\bar{u}_i/\partial x_j + \partial\bar{u}_j/\partial x_i)$:

$$R_{ij} = -2\mu_{SGS}\bar{S}_{ij} + \frac{1}{3}R_{ii}\delta_{ij} = -\mu_{SGS}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) + \frac{1}{3}R_{ii}\delta_{ij} \tag{3.93}$$

The constant of proportionality is the dynamic SGS viscosity μ_{SGS}, which has dimensions Pa s. The term $\frac{1}{3}R_{ii}\delta_{ij}$ on the right hand side of equation (3.93) performs the same function as the term $-\frac{2}{3}\rho k\delta_{ij}$ in equation (3.33): it ensures that the sum of the modelled normal SGS stresses is equal to the kinetic energy of the SGS eddies. In much of the LES research literature the above model is used along with approximate forms of the Leonard stresses L_{ij} and cross-stresses C_{ij} for the particular filtering function applied in the work.

Meinke and Krause (in Peyret and Krause, 2000) review applications of finite volume/LES to complex, industrially relevant CFD computations. These authors note that, in spite of the different nature of the Leonard stresses and cross-stresses, they are lumped together with the LES Reynolds stresses in the current versions of the **finite volume method**. The whole stress τ_{ij} is modelled as a single entity by means of a single **SGS turbulence model**:

$$\tau_{ij} = -2\mu_{SGS}\bar{S}_{ij} + \frac{1}{3}\tau_{ii}\delta_{ij} = -\mu_{SGS}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) + \frac{1}{3}\tau_{ii}\delta_{ij} \tag{3.94}$$

The Smagorinsky–Lilly SGS model builds on Prandtl's mixing length model (3.39) and assumes that we can define a kinematic SGS viscosity ν_{SGS} (dimensions m^2/s), which can be described in terms of one length scale and one velocity scale and is related to the dynamic SGS viscosity by $\nu_{SGS} = \mu_{SGS}/\rho$. Since the size of the SGS eddies is determined by the details of the filtering function, the obvious choice for the length scale is the filter cutoff width Δ. As in the mixing length model, the velocity scale is expressed as the product of the length scale Δ and the average strain rate of the resolved flow $\Delta \times |\bar{S}|$, where $|\bar{S}| = \sqrt{2\bar{S}_{ij}\bar{S}_{ij}}$. Thus, the **SGS viscosity** is evaluated as follows:

$$\mu_{SGS} = \rho(C_{SGS}\Delta)^2|\bar{S}| = \rho(C_{SGS}\Delta)^2\sqrt{2\bar{S}_{ij}\bar{S}_{ij}} \tag{3.95}$$

where C_{SGS} = constant

and $\bar{S}_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)$

Lilly (1966, 1967) presented a theoretical analysis of the decay rates of isotropic turbulent eddies in the inertial subrange of the energy spectrum, which suggests values of C_{SGS} between 0.17 and 0.21. Rogallo and Moin (1984) reviewed work by other authors suggesting values of C_{SGS} = 0.19–0.24 for results across a range of grids and filter functions. They also quoted early LES computations by Deardorff (1970) of turbulent channel flow, which has strongly anisotropic turbulence, particularly in the near-wall regions. This work established that the above values caused excessive damping and suggested that C_{SGS} = 0.1 is most appropriate for this type of internal flow calculation.

The difference in C_{SGS} values is attributable to the effect of the mean flow strain or shear. This gave an early indication that the behaviour of the small eddies is not as universal as was surmised at first and that successful LES turbulence modelling might require case-by-case adjustment of C_{SGS} or a more sophisticated approach.

3.8.3 Higher-order SGS models

A model of the SGS Reynolds stresses based on the Boussinesq eddy viscosity hypothesis assumes that changes in the resolved flow take place sufficiently slowly that the SGS eddies can adjust themselves instantaneously to the rate of strain of the resolved flow field. An alternative strategy to case-by-case tuning of the constant C_{SGS} is to use the ideas of RANS turbulence modelling to make an allowance for transport effects. We keep the filter cutoff width Δ as the characteristic length scale of the SGS eddies, but replace the velocity scale $\Delta \times |\bar{S}|$ by one that is more representative of the velocity of the SGS eddies. For this we choose the square root of the SGS turbulent kinetic energy $\sqrt{k_{SGS}}$. Thus,

$$\boxed{\mu_{SGS} = \rho C'_{SGS} \Delta \sqrt{k_{SGS}}} \tag{3.96}$$

where C'_{SGS} = constant

To account for the effects of convection, diffusion, production and destruction on the SGS velocity scale we solve a transport equation to determine the distribution of k_{SGS}:

$$\boxed{\frac{\partial(\rho k_{SGS})}{\partial t} + \text{div}(\rho k_{SGS}\bar{\mathbf{u}}) = \text{div}\left(\frac{\mu_{SGS}}{\sigma_k}\,\text{grad}(k_{SGS})\right) + 2\mu_{SGS}\bar{S}_{ij}\,.\,\bar{S}_{ij} - \rho\varepsilon_{SGS}} \tag{3.97}$$

Dimensional analysis shows that the rate of dissipation ε_{SGS} of SGS turbulent kinetic energy is related to the length and velocity scales as follows:

$$\varepsilon_{SGS} = C_\varepsilon \frac{k_{SGS}^{3/2}}{\Delta} \tag{3.98}$$

where C_ε = constant

This is the LES equivalent of a one-equation RANS turbulence model, such as the one used in the two-layer k–ε model for the viscous-dominated near-wall region. Schumann (1975) successfully used such a model to compute turbulent flows in two-dimensional channels and annuli. In a more recent study Fureby *et al*. (1997) have carried out LES computations of homogeneous isotropic turbulence, which has revived interest in this model, leading to its implementation in the commercial CFD code STAR-CD.

The above SGS models are all based on the Boussinesq assumption of a constant SGS eddy viscosity to link SGS stresses and resolved-flow strain rates. Challenging this isotropic eddy viscosity assumption naturally leads to the LES equivalent of the Reynolds stress model. Deardorff (1973) used this model in computations of the atmospheric boundary layer, where the filter cutoff width must be chosen so large that the unresolved turbulent eddies are anisotropic and the eddy viscosity assumption becomes inaccurate.

3.8.4 Advanced SGS models

The Smagorinsky model is purely dissipative: the direction of energy flow is exclusively *from eddies at the resolved scales towards the sub-grid scales*. Leslie and Quarini (1979) have shown that the gross energy flow in this direction is actually larger and offset by 30% backscatter – energy transfer in reverse direction from SGS eddies to larger, resolved scales. Furthermore, analysis of results from direct numerical simulation (DNS) by Clark *et al*. (1979) and McMillan and Ferziger (1979) revealed that the correlation between the *actual* SGS stresses (as computed by accurate DNS) and the *modelled* SGS stresses using the Smagorinsky–Lilly model is not particularly strong. These authors came to the conclusion that the SGS stresses should not be taken as proportional to the strain rate of the whole resolved flow field, but, in recognition of the actual energy cascade processes (see section 3.1), should be estimated from the strain rate of the smallest resolved eddies. Bardina *et al*. (1980) proposed a method to compute local values of C_{SGS} based on the application of two filtering operations, taking the SGS stresses to be proportional to the stresses due to eddies at the smallest resolved scale. They proposed

$$\tau_{ij} = \rho C'(\overline{\bar{u}_i \bar{u}_j} - \bar{\bar{u}}_i \bar{\bar{u}}_j)$$

where C' is an adjustable constant and the factor in the brackets can be evaluated from twice-filtered resolved flow field information. The correlation between the *actual* SGS stresses as computed with a DNS and the *modelled* SGS stresses was found to be much improved, but the appearance of negative viscosities generated stability problems. They proposed adding a damping term with the form of the Smagorinsky model (3.94)–(3.95) to stabilise the calculations, which yields a **mixed model**:

$$\tau_{ij} = \rho C'(\overline{\bar{u}_i \bar{u}_j} - \bar{\bar{u}}_i \bar{\bar{u}}_j) - 2\rho C_{SGS}^2 \Delta^2 |\bar{S}| \bar{S}_{ij} \tag{3.99}$$

The value of the constant C' depends on the cutoff width used for the second filtering operation, but is always close to unity.

Germano (1986) proposed a different decomposition of the turbulent stresses. This formed the basis of the **dynamic SGS model** (Germano *et al*., 1991) for the computation of local values of C_{SGS}. In Germano *et al*.'s decomposition of turbulent stresses the difference of the SGS stresses for two different filtering operations with cutoff widths Δ_1 and Δ_2, respectively, can be evaluated from resolved flow data:

$$\tau_{ij}^{(2)} - \tau_{ij}^{(1)} = \rho L_{ij} \equiv \rho\, \overline{\bar{u}_i \bar{u}_j} - \rho\, \bar{\bar{u}}_i \bar{\bar{u}}_j \tag{3.100}$$

The bracketed superscripts (1) and (2) indicate filtering at cutoff widths Δ_1 and Δ_2.

The SGS stresses are modelled using Smagorinsky's model (3.94)–(3.95) assuming that the constant C_{SGS} is the same for both filtering operations. It can be shown that this yields:

$$L_{ij} - \frac{1}{3} L_{kk} \delta_{ij} = C_{SGS}^2 M_{ij} \tag{3.101a}$$

$$\text{with } M_{ij} = -2\Delta_2^2 |\bar{\bar{S}}| \bar{\bar{S}}_{ij} + 2\Delta_1^2 \overline{|\bar{S}| \bar{S}_{ij}} \tag{3.101b}$$

Lilly (1992) suggested a least-squares approach to evaluate local values of C_{SGS}:

$$C_{SGS}^2 = \frac{\langle L_{ij} M_{ij} \rangle}{\langle M_{ij} M_{ij} \rangle} \tag{3.102}$$

The angular brackets $\langle\ \rangle$ indicate an averaging procedure. As Bardina *et al.* (1980) before them, Germano *et al.* (1991) found that the dynamic SGS model yielded highly variable eddy viscosity fields including regions with negative values. This problem was resolved by averaging: for problems with homogeneous directions (e.g. two-dimensional planar flows) the averaging takes place over the homogeneous direction; in complex flows an average over a small time interval is used.

Germano (in Peyret and Krause, 2000) reviewed other formulations for the dynamic calculation of the SGS eddy viscosity. The dynamic model and other advanced SGS models are reviewed in Lesieur and Métais (1996) and Meneveau and Katz (2000). The interested reader is referred to these publications for further material in this area.

3.8.5 Initial and boundary conditions for LES

In LES computations the unsteady Navier–Stokes equations are solved, so suitable initial and boundary conditions must be supplied to generate a well-posed problem.

Initial conditions

For steady flows the initial state of the flow only determines the length of time required to reach the steady state, and it is usually adequate to specify an initial field that conserves mass with superimposed Gaussian random fluctuations with the correct turbulence level or spectral content. If the development of a time-dependent flow depends on its initial state it is necessary to specify it more accurately using data from other sources (DNS or experiments).

Solid walls

The no-slip condition is used if the LES filtered Navier–Stokes equations are integrated to the wall, which requires fine grids with near-wall grid points $y^+ \le 1$. For high Reynolds number flows with thin boundary layers it is necessary to economise on computing resources by means of graded non-uniform meshes. As an alternative it is possible to use wall functions. Schumann (1975) proposed a model that takes the fluctuating shear stress to be in phase with the fluctuating velocity parallel to the wall and links the shear stress to the instantaneous velocity through logarithmic wall functions of the same type as equation (3.49) used in the RANS k–ε model and RSM. Moin (2002) reviewed an advanced method of dynamically computing von Karman's proportionality constant κ in his near-wall RANS mixing length model by matching the values of the RANS and LES eddy viscosities at matching points. This avoids excessive shear stress predictions associated with the standard value $\kappa = 0.41$.

Inflow boundaries

Inflow boundary conditions are very challenging since the inlet flow properties are convected downstream, and inaccurate specification of the inflow

boundary condition can strongly affect simulation quality. The simplest method is to specify measured mean velocity distributions and to superimpose Gaussian random perturbations with the correct turbulence intensity, but this ignores the cross-correlations between velocity components (Reynolds stresses) and two-point correlations (i.e. spatial coherence) in real turbulent flows. Distortions of turbulence properties can take considerable settling distance before the mean flow reaches equilibrium with the turbulence properties, and the settling distance is problem dependent. Several alternatives are available:

1 Represent the inlet flow with the correct geometry using a RANS turbulence model. The commonest method is to perform an unsteady flow calculation with the RSM to obtain estimates of all the Reynolds stresses at the inlet plane and impose these by maintaining correct values of the relevant autocorrelations and cross-correlations during the generation of the Gaussian random perturbations.
2 Extend the computational domain further upstream and use a turbulence-free inflow (by developing the flow from a large reservoir). This requires a long upstream distance, typically of the order of 50 hydraulic diameters, until a fully developed flow is reached, but is feasible if an inlet flow with thin boundary layers is required.
3 Specify a fully developed inlet profile as the starting point for internal flows in complex geometry. Such profiles can be economically computed from an auxiliary LES computation with streamwise periodic boundaries (see below).
4 Specify a precise inlet profile with prescribed shear stress, momentum thickness and boundary layer thickness. Lund *et al.* (1998) proposed a technique to extract inlet profiles for developing boundary layers from auxiliary LES computations. Other methods with this objective have been developed by Klein *et al.* (2003) and Ferrante and Elgobashi (2004). The former is based on digital filtering of random data and the specification of length scales in each co-ordinate direction to generate two-point correlations. The latter proposes a refinement of the Lund *et al.* procedure to ensure that the correct spectral energy distribution is reproduced across the wavenumbers. Both algorithms are reported to reduce the settling length between the inflow boundary and the location in the computational domain of the actual LES calculations where the turbulence reaches equilibrium with the mean flow.

Outflow boundaries

Outflow boundary conditions are less troublesome. The familiar zero gradient boundary condition is used for the mean flow, and the fluctuating properties are extrapolated by means of a so-called convective boundary condition:

$$\frac{\partial \phi}{\partial t} + \bar{u}_n \frac{\partial \phi}{\partial n} = 0$$

Periodic boundary conditions

All LES and DNS calculations are three-dimensional because turbulence is three-dimensional. Periodic boundary conditions are particularly useful in directions where the mean flow is homogeneous (e.g. the z-direction in

two-dimensional planar flow). All properties are set to be equal at equivalent points on pairs of periodic boundaries. The distance between the two periodic boundaries must be such that two-point correlations are zero for all points on a pair of periodic boundaries. This means that the distance should be chosen to be at least twice the size of the largest eddies so that the effect of one boundary on the other is minimal.

3.8.6 LES applications in flows with complex geometry

Considerable effort has been made in the research community to develop robust LES methods for general-purpose CFD computations involving complex geometry. We briefly summarise some of the issues that have been addressed in the recent literature.

Non-uniform grids are preferable in flows with solid boundaries to resolve the rapid changes in the near-wall region. However, this would require different filter cutoff width in the core flow and near-wall regions. Suppose we adjust the cutoff width of a filter to give the correct separation between large and small eddy scales in the core flow. This cutoff width would be inappropriate for the near-wall region, where the size of turbulence eddies is restricted by the presence of the solid boundary. Here the chosen filter cutoff width would be too large and would cause anisotropic, energetic near-wall eddies to be included in the SGS scales. Equation (3.86) states that the cutoff width for three-dimensional computations should be taken as the cube root of the control volume size. In non-uniform grids the cutoff width would vary along with the control volume size. Scotti *et al*. (1993) show that Smagorinsky's constant C_{SGS} should be corrected to take into account grid anisotropy in the three dimensions:

$$C_{SGS} = 0.16 \times \cosh\sqrt{\tfrac{4}{27}[(\ln a_1)^2 - \ln a_1 \times \ln a_2 + (\ln a_2)^2]}$$

where the grid-anisotropy factors are given by $a_1 = \Delta x/\Delta z$ and $a_2 = \Delta y/\Delta z$.

In finite volume applications the filter cutoff width $\Delta = \sqrt[3]{\Delta x \Delta y \Delta z}$ is necessarily close to the mesh size. There is a price to pay because this choice of cutoff width blurs the distinction between the effects of the SGS eddies and the numerical errors associated with the discretisation of the equations on the grid. The SGS stresses will be similar in magnitude to the numerical truncation errors. Unless careful attention is paid to the control of **numerical errors** they may even swamp the SGS stresses. Upwind differencing (see Chapter 5), which was standard practice in early CFD computations with RANS turbulence modelling, is far too diffusive and generates large truncation errors. Second-order or higher-order discretisation techniques are needed. Moin (2002) reported the performance of a robust and non-dissipative discretisation method for unstructured grids in simulations of a gas turbine combustor.

A uniform cutoff width was assumed in the development of the LES equations (3.87) and (3.89a–c) to ensure that filtering commutes with time and space derivatives. This is not the case when the cutoff width is non-uniform. Ghosal and Moin (1995) show that the non-commutativity errors associated with **non-uniform cutoff width** can be of similar magnitude to the Leonard stresses and truncation errors. They propose a method based on transformation of the co-ordinate system to control this error. Further methods to minimise this problem have since been proposed: Vasilyev *et al*.

(1998) developed a class of discrete commutative filters for non-uniform structured grids, which was extended by Marsden *et al*. (2002) to unstructured grids for use in conjunction with second-order accurate discretisation schemes.

3.8.7 General comments on performance of LES

It is the main task of turbulence modelling to develop computational procedures of sufficient accuracy and generality for engineers to predict the Reynolds stresses and the scalar transport terms. The inherent unsteady nature of LES suggests that the computational requirements should be much larger than those of classical turbulence models. This is indeed the case when LES is compared with two-equation models such as the k–ε and k–ω models. However, RSMs require the solution of seven additional PDEs, and Ferziger (1977) noted that LES may only need about twice the computer resources compared with RSM for the same calculation. With such modest differences in computational requirements the focus switches to the achievable solution accuracy and the ability of the LES to resolve certain time-dependent features 'for free'. Post-processing of LES results yields information relating to the mean flow and statistics of the resolved fluctuations. The latter are unique to LES, and Moin as well as Meinke and Krause (both in Peyret and Krause, 2000) gave examples of flows where persistent large-scale vortices have a substantial influence on flow development, e.g. vortex shedding behind bluff bodies, flows in diffusing passages, flows in pipe bends and tumble swirl in engine combustion chambers. The ability to obtain fluctuating pressure fields from LES output has also led to aeroacoustic applications for the prediction of noise from jets and other high-speed flows.

As an illustration of the most advanced LES capability we show results from Moin (2002) for a gas turbine. Figure 3.17 shows a detail of the combustor geometry and the computational grid, which is unstructured to model all the details in this very complex geometry. Figure 3.18 shows contours of instantaneous velocity magnitude on a mid-section plane and on four further perpendicular cross-sections as indicated on the diagram. The physics of the turbulent flow is also highly complex, involving combustion, swirl, dilution jets etc. Flow instabilities have serious consequences for combustion, and the information generated by LES calculations is uniquely applicable to the development of this technology.

Figure 3.17 LES computations on Pratt & Whitney gas turbine – detail of combustor geometry and computational grid
Source: Moin (2002)

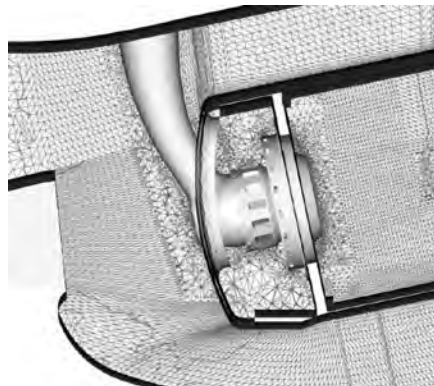
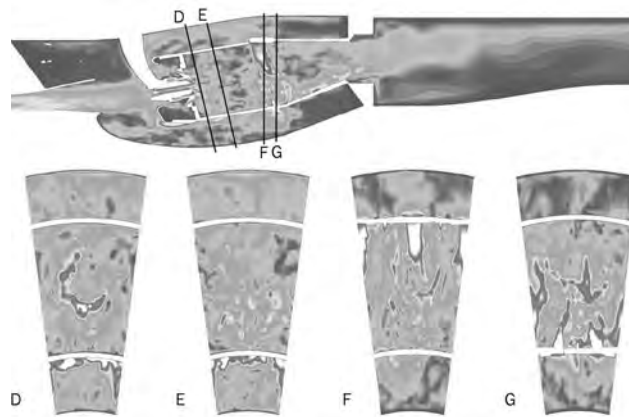



Figure 3.18 LES computations on Pratt & Whitney gas turbine – instantaneous contours of velocity magnitude on sectional planes
Source: Moin (2002)

LES has been around since the 1960s, but sufficiently powerful computing resources to consider application to industrially relevant problems have only recently become available. Inclusion of LES in commercial CFD is even more recent, so the range of validation experience is limited. Most code vendors usually state that care must be taken with the interpretation of results generated with their LES models. Furthermore, it should be noted that the methodology for the treatment of non-commutativity effects in non-uniform and unstructured grids is comparatively recent, as are treatments for compressible flow and turbulent scalar fluctuations. This research does not yet appear to have been incorporated in finite volume/LES codes. Geurts and Leonard (2005) give a survey of the main issues that need to be addressed to control error sources and generate robust LES methodology for application to industrially relevant complex flows. It is likely that the pace of developments will increase as computing resources become more powerful and as the CFD user community becomes more aware of the advantages of the LES approach to turbulence modelling.

3.9 Direct numerical simulation

The instantaneous continuity and Navier–Stokes equations (3.23) and (3.24a–c) for an incompressible turbulent flow form a closed set of four equations with four unknowns u, v, w and p. **Direct numerical simulation (DNS)** of turbulent flow takes this set of equations as a starting point and develops a transient solution on a sufficiently fine spatial mesh with sufficiently small time steps to resolve even the smallest turbulent eddies and the fastest fluctuations.

Reynolds (in Lumley, 1989) and Moin and Mahesh (1998) listed the potential benefits of DNSs:

- Precise details of turbulence parameters, their transport and budgets at any point in the flow can be calculated with DNS. These are useful for the development and validation of new turbulence models. Refereed databases giving free access to DNS results have started to emerge (e.g. ERCOFTAC, http://ercoftac.mech.surrey.ac.uk/dns/homepage.html; Turbulence and Heat Transfer Lab of the University of Tokyo, http://www.thtlab.t.u-tokyo.ac.jp; the University of Manchester, http://cfd.me.umist.ac.uk/ercoftac).

- Instantaneous results can be generated that are not measurable with instrumentation, and instantaneous turbulence structures can be visualised and probed. For example, pressure–strain correlation terms in RSM turbulence models cannot be measured, but accurate values can be computed from DNSs.
- Advanced experimental techniques can be tested and evaluated in DNS flow fields. Reynolds (in Lumley, 1989) noted that DNS has been used to calibrate hot-wire anemometry probes in near-wall turbulence.
- Fundamental turbulence research on virtual flow fields that cannot occur in reality, e.g. by including or excluding individual aspects of flow physics. Moin and Mahesh (1998) listed some examples: shear-free boundary layers developing on walls at rest with respect to the free stream, effect of initial conditions on the development of self-similar turbulent wakes, the study of the fundamentals of reacting flows (strain rates of flamelets and distortion of mixing surfaces).

On the downside we note that direct solution of the flow equations is very difficult because of the wide range of length and time scales caused by the appearance of eddies in a turbulent flow. In section 3.1 we considered order-of-magnitude estimates of the range of scales present in a turbulent flow and found that the ratio of smallest to largest length scales varied in proportion to $Re^{3/4}$. To resolve the smallest and largest turbulence length scales a direct simulation of a turbulent flow with a modest Reynolds number of 10^4 would require of the order of 10^3 points in each co-ordinate direction. Thus, since turbulent flows are inherently three-dimensional, we would need computing meshes with 10^9 grid points ($N \cong Re^{9/4}$) to describe processes at all length scales. Furthermore, the ratio of smallest to largest time scales varies as $Re^{1/2}$, so at $Re = 10^4$ we would need to run a simulation for at least 100 time steps. In practice, a larger number of time steps would be needed to ensure the passage of several of the largest eddies in order to obtain meaningful time-average flow results and turbulence statistics.

Speziale (1991) estimated that the direct simulation of a turbulent pipe flow at a Reynolds number of 500 000 requires a computer which is 10 million times faster than a (then) current generation Cray supercomputer. Moin and Kim (1997) estimated computing times of 100 hours to 300 years for turbulent flows at Reynolds numbers in the range 10^4 to 10^6 based on high-performance computer speeds of 150 Mflops available at that time. This confirmed that it started to become possible to compute interesting turbulent flows with DNS based on the unsteady Navier–Stokes equations. Advanced supercomputers at present (2006) have processor speeds of the order 1–10 Tflops. If the performance scaling can be maintained across such a wide range of speeds, this would reduce computing times to minutes or hours. We briefly review progress in this rapidly growing area of turbulence research.

3.9.1 Numerical issues in DNS

It is of course beyond the scope of this introduction to go into the details of the methods used for DNS, but it is worth touching on the specific requirements for this type of computation. The review by Moin and Mahesh (1998) highlights the following issues being tackled in the DNS research literature.

Spatial discretisation

The first DNS simulations were performed with **spectral methods** (Orszag and Patterson, 1972). These are based on Fourier series decomposition in periodic directions and Chebyshev polynomial expansions in directions with solid walls. The methods are economical and have high convergence rates, but they are difficult to apply in complex geometry. Nevertheless, they are still widely used in research on transitional flows and turbulent flows with simple geometries: some recent applications are flow-induced vibrations (Evangelinos *et al*., 2000), strained two-dimensional wake flow (Rogers, 2002) and transition in rotor–stator cavity flow (Serre *et al*., 2002, 2004). Early recognition of the limitations of spectral methods led to the development of **spectral element methods** (Orszag and Patera, 1984; Patera, 1986). These combine the geometric flexibility of the finite element method with the good convergence properties of the spectral method. These methods have been developed for complex turbulent flows by Karniadakis and co-workers (e.g. Karniadakis, 1989, 1990).

Higher-order finite difference methods (Moin, 1991) are now widely used for problems with more complex geometry. Particular attention needs to be paid to the design of the spatial and temporal discretisation schemes to ensure that the method is stable and to make sure that numerical dissipation does not swamp turbulent eddy dissipation. A sample of recent work illustrates the range of applications: turbulent flow in a pipe rotating about its axis (Orlandi and Fatica, 1997), flow around square cylinders (Tamura *et al*., 1998), plumes (Jiang and Luo, 2000) and diffusion flames (Luo *et al*., 2005).

Spatial resolution

Above we have noted that the spatial mesh for DNS is determined at one end by the largest geometrical features that need to be resolved and at the other end by the finest turbulence scales that are generated. Research has shown that the grid point requirement $N \propto Re^{9/4}$ can be somewhat relaxed, because most of the dissipation actually takes place at scales that are substantially larger than of the order of the Kolmogorov length scale η, say 5η–15η (Moin and Mahesh, 1998). As long as the bulk of the dissipation process is adequately represented, the number of grid cells can be reduced. In typical finite difference calculations reduction by a factor of around 100 is possible without significant loss of accuracy.

Temporal discretisation

There is a wide range of time scales in a turbulent flow, so the system of equations is stiff. Implicit time advancement and large time steps are routinely used for stiff systems in general-purpose CFD, but these are unsuitable in DNS because complete time resolution is needed to describe the energy dissipation process accurately. Specially designed implicit and explicit methods have been developed to ensure time accuracy and stability (see e.g. Verstappen and Veldman, 1997).

Temporal resolution

Reynolds (in Lumley, 1989) noted that it is essential to have accurate time resolution of all the scales of turbulent motion. The time steps must be

adjusted so that fluid particles do not move more than one mesh spacing. Moin and Mahesh (1998) demonstrated the strong influence of time step size on small-scale amplitude and phase error.

Initial and boundary conditions

Issues relating to initial and boundary conditions are similar to those in LES. The reader is referred to section 3.8.5 for a relevant discussion.

3.9.2 Some achievements of DNS

Early work on transitional flows is reviewed in Kleiser and Zang (1991). Since the paper by Orszag and Patterson (1972) a range of turbulent incompressible flows of fundamental importance have been investigated. We list the most important studies and refer the interested reader to the review paper by Moin and Mahesh (1998) for further details: homogeneous turbulence with mean strain, free shear layers, fully developed channel flow, curved channel flow, channel flow with riblets, channel flow with heat transfer, rotating channel flow, channel flow with transverse curvature, flow over a backward-facing step, flat plate boundary layer separation.

More recently the DNS methodology has been extended to compressible flows: homogeneous compressible turbulence, isotropic and sheared compressible turbulence, compressible channel flow, compressible turbulent boundary layer, high-speed compressible turbulent mixing layer.

During the first two decades after the study by Orszag and Patterson the resources required for DNS calculations were only available to a handful of groups across the globe. Since the 1990s, however, high-performance computing has become much more widely available, and the technique is within reach of a much larger number of turbulence researchers with more diverse interests including fundamental aspects of flows with multi-physics: gas–liquid turbulent flows, particle-laden flows (Elghobashi and Truesdell, 1993) and reacting flows (Poinsot *et al.*, 1993). The unique ability of DNS to generate accurate flow fields has led to the development of the new field of computational aeroacoustics (reviewed in Tam, 1995; Lele, 1997).

Although most DNS computations are at comparatively low Reynolds numbers (e.g. Hoarau *et al.*, 2003), predictions in the 1960s and 1970s that DNS would *never* be a realistic proposition for turbulent flows of relevance to engineering may have been unduly pessimistic. Much effort will be focused on speed and stability improvements of the basic numerical algorithms (see e.g. Verstappen and Veldman, 1997) as well as the development of methods to take best advantage of future high-performance computer architectures. Given the potential benefits of DNS results it is likely that rapid growth of interest in this area is set to continue.

3.10 Summary

This chapter provides a first glimpse of turbulent flows and of the practice of turbulence modelling in CFD. Turbulence is a phenomenon of great complexity and has challenged leading theoreticians for over a hundred years. The flow fluctuations associated with turbulence give rise to additional transfer of momentum, heat and mass. These changes to the flow character can be

favourable (efficient mixing) or detrimental (high energy losses) depending on one's point of view.

Engineers are mainly interested in the prediction of mean flow behaviour, but turbulence cannot be ignored, because the fluctuations give rise to the extra Reynolds stresses on the mean flow. These extra stresses must be modelled in industrial CFD. What makes the prediction of the effects of turbulence so difficult is the wide range of length and time scales of motion even in flows with very simple boundary conditions. It should therefore be considered as truly remarkable that RANS turbulence models, such as the k–ε models, succeed in expressing the main features of many turbulent flows by means of one length scale and one time scale defining variable. The standard k–ε model is valued for its robustness, and is still widely preferred in industrial internal flow computations. The k–ω model and Spalart–Allmaras model have become established as the leading RANS turbulence models for aerospace applications. Many experts argue that the RSM is the only viable way forward towards a truly general-purpose classical turbulence model, but recent advances in the area of non-linear k–ε models are very likely to reinvigorate research on two-equation turbulence models. As a cautionary note, Leschziner (in Peyret and Krause, 2000) observes that performance improvements of new RANS turbulence models have not been uniform: in some cases the cubic k–ε model performs as well as the RSM, whereas in other cases it is not discernibly better than the standard k–ε model, so the verdict on these models is still open.

Large eddy simulation (LES) requires substantial computing resources, and the technique needs further research and development before it can be applied as an industrial general-purpose tool in flows with complex geometry. However, it is already recognised that valuable information can be obtained from LES computations in simple flows by generating turbulence properties that cannot be measured in the laboratory due to the absence of suitable experimental techniques. Hence, as a research tool LES will increasingly be used to guide the development of classical models through comparative studies. Several commercial CFD codes now contain basic LES capability, and these are likely to see more widespread industrial applications in flows where large-scale time-dependent flow features (vortex shedding, swirl etc.) play a role. The emergence of high-performance computing resources based around Linux PC clusters is likely to accelerate this trend.

Although the resulting mathematical expressions of turbulence models may be quite complicated it should never be forgotten that they all contain **adjustable constants** that need to be determined as best-fit values from experimental data that contain experimental uncertainties. Every engineer is aware of the dangers of extrapolating an empirical model beyond its data range. The same risks occur when (ab)using turbulence models in this fashion. CFD calculations of 'new' turbulent flows should never be accepted without validation against high-quality data. The source can be experiments, but increasingly the data that can be generated by means of numerical experiments with DNS are being used as benchmarks. DNS is likely to play an increasingly important role in turbulence research in the near future.

Chapter four The finite volume method for diffusion problems

4.1 Introduction

The nature of the transport equations governing fluid flow and heat transfer and the formal control volume integration were described in Chapter 2. Here we develop the numerical method based on this integration, the **finite volume (or control volume) method**, by considering the simplest transport process of all: pure diffusion in the steady state. The governing equation of steady diffusion can easily be derived from the general transport equation (2.39) for property ϕ by deleting the transient and convective terms. This gives

$$\mathrm{div}(\Gamma\ \mathrm{grad}\ \phi) + S_\phi = 0 \tag{4.1}$$

The control volume integration, which forms the key step of the finite volume method that distinguishes it from all other CFD techniques, yields the following form:

$$\int_{CV} \mathrm{div}(\Gamma\ \mathrm{grad}\ \phi)\mathrm{d}V + \int_{CV} S_\phi \mathrm{d}V = \int_A \mathbf{n}\ .\ (\Gamma\ \mathrm{grad}\ \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V = 0 \tag{4.2}$$

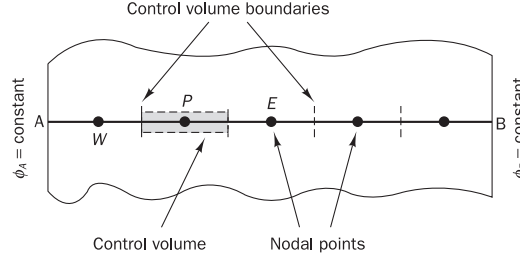
By working with the one-dimensional steady state diffusion equation, the approximation techniques that are needed to obtain the so-called discretised equations are introduced. Later the method is extended to two- and three-dimensional diffusion problems. Application of the method to simple one-dimensional steady state heat transfer problems is illustrated through a series of worked examples, and the accuracy of the method is gauged by comparing numerical results with analytical solutions.

4.2 Finite volume method for one-dimensional steady state diffusion

Consider the steady state diffusion of a property ϕ in a one-dimensional domain defined in Figure 4.1. The process is governed by

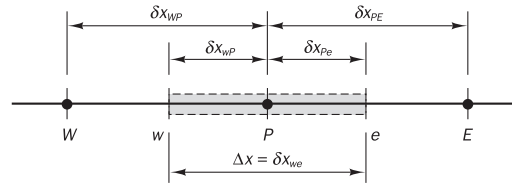
$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\phi}{\mathrm{d}x}\right) + S = 0 \tag{4.3}$$

where Γ is the diffusion coefficient and S is the source term. Boundary values of ϕ at points A and B are prescribed. An example of this type of process, one-dimensional heat conduction in a rod, is studied in detail in section 4.3.

Figure 4.1

Step 1: Grid generation

The first step in the finite volume method is to divide the domain into discrete control volumes. Let us place a number of nodal points in the space between A and B. The boundaries (or faces) of control volumes are positioned mid-way between adjacent nodes. Thus each node is surrounded by a control volume or cell. It is common practice to set up control volumes near the edge of the domain in such a way that the physical boundaries coincide with the control volume boundaries.

At this point it is appropriate to establish a system of notation that can be used in future developments. The usual convention of CFD methods is shown in Figure 4.2.

Figure 4.2

A general nodal point is identified by P and its neighbours in a one-dimensional geometry, the nodes to the west and east, are identified by W and E respectively. The west side face of the control volume is referred to by w and the east side control volume face by e. The distances between the nodes W and P, and between nodes P and E, are identified by δx_{WP} and δx_{PE} respectively. Similarly distances between face w and point P and between P and face e are denoted by δx_{wP} and δx_{Pe} respectively. Figure 4.2 shows that the control volume width is $\Delta x = \delta x_{we}$.

Step 2: Discretisation

The key step of the finite volume method is the integration of the governing equation (or equations) over a control volume to yield a discretised equation at its nodal point P. For the control volume defined above this gives

$$\int_{\Delta V} \frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma \frac{\mathrm{d}\phi}{\mathrm{d}x}\right)\mathrm{d}V + \int_{\Delta V} S\,\mathrm{d}V = \left(\Gamma A \frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_e - \left(\Gamma A \frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_w + \bar{S}\Delta V = 0 \tag{4.4}$$

Here A is the cross-sectional area of the control volume face, ΔV is the volume and $\bar{S}$ is the average value of source S over the control volume. It is

a very attractive feature of the finite volume method that the discretised equation has a clear physical interpretation. Equation (4.4) states that the diffusive flux of ϕ leaving the east face minus the diffusive flux of ϕ entering the west face is equal to the generation of ϕ, i.e. it constitutes a balance equation for ϕ over the control volume.

In order to derive useful forms of the discretised equations, the interface diffusion coefficient Γ and the gradient $\mathrm{d}\phi/\mathrm{d}x$ at east (e) and west (w) are required. Following well-established practice, the values of the property ϕ and the diffusion coefficient are defined and evaluated at nodal points. To calculate gradients (and hence fluxes) at the control volume faces an approximate distribution of properties between nodal points is used. Linear approximations seem to be the obvious and simplest way of calculating interface values and the gradients. This practice is called central differencing (see Appendix A). In a uniform grid linearly interpolated values for Γ_w and Γ_e are given by

$$\Gamma_w = \frac{\Gamma_W + \Gamma_P}{2} \tag{4.5a}$$

$$\Gamma_e = \frac{\Gamma_P + \Gamma_E}{2} \tag{4.5b}$$

And the diffusive flux terms are evaluated as

$$\left(\Gamma A \frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_e = \Gamma_e A_e \left(\frac{\phi_E - \phi_P}{\delta x_{PE}}\right) \tag{4.6}$$

$$\left(\Gamma A \frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_w = \Gamma_w A_w \left(\frac{\phi_P - \phi_W}{\delta x_{WP}}\right) \tag{4.7}$$

In practical situations, as illustrated later, the source term S may be a function of the dependent variable. In such cases the finite volume method approximates the source term by means of a linear form:

$$\bar{S}\Delta V = S_u + S_p \phi_P \tag{4.8}$$

Substitution of equations (4.6), (4.7) and (4.8) into equation (4.4) gives

$$\Gamma_e A_e \left(\frac{\phi_E - \phi_P}{\delta x_{PE}}\right) - \Gamma_w A_w \left(\frac{\phi_P - \phi_W}{\delta x_{WP}}\right) + (S_u + S_p \phi_P) = 0 \tag{4.9}$$

This can be rearranged as

$$\boxed{\left(\frac{\Gamma_e}{\delta x_{PE}} A_e + \frac{\Gamma_w}{\delta x_{WP}} A_w - S_p\right)\phi_P = \left(\frac{\Gamma_w}{\delta x_{WP}} A_w\right)\phi_W + \left(\frac{\Gamma_e}{\delta x_{PE}} A_e\right)\phi_E + S_u} \tag{4.10}$$

Identifying the coefficients of ϕ_W and ϕ_E in equation (4.10) as a_W and a_E, and the coefficient of ϕ_P as a_P, the above equation can be written as

$$\boxed{a_P \phi_P = a_W \phi_W + a_E \phi_E + S_u} \tag{4.11}$$

where

a_W	a_E	a_P
$\frac{\Gamma_w}{\delta x_{WP}} A_w$	$\frac{\Gamma_e}{\delta x_{PE}} A_e$	$a_W + a_E - S_P$

The values of S_u and S_p can be obtained from the source model (4.8): $\bar{S}\Delta V = S_u + S_p\phi_P$. Equations (4.11) and (4.8) represent the discretised form of equation (4.1). This type of discretised equation is central to all further developments.

Step 3: Solution of equations

Discretised equations of the form (4.11) must be set up at each of the nodal points in order to solve a problem. For control volumes that are adjacent to the domain boundaries the general discretised equation (4.11) is modified to incorporate boundary conditions. The resulting system of linear algebraic equations is then solved to obtain the distribution of the property ϕ at nodal points. Any suitable matrix solution technique may be enlisted for this task. In Chapter 7 we describe matrix solution methods that are specially designed for CFD procedures. The techniques of dealing with different types of boundary conditions will be examined in detail in Chapter 9.

4.3 Worked examples: one-dimensional steady state diffusion

The application of the finite volume method to the solution of simple diffusion problems involving conductive heat transfer is presented in this section. The equation governing one-dimensional steady state conductive heat transfer is

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right) + S = 0 \tag{4.12}$$

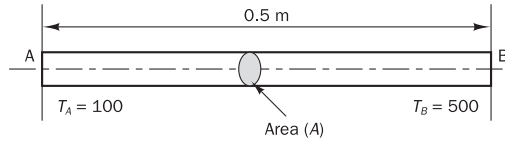
where thermal conductivity k takes the place of Γ in equation (4.3) and the dependent variable is temperature T. The source term can, for example, be heat generation due to an electrical current passing through the rod. Incorporation of boundary conditions as well as the treatment of source terms will be introduced by means of three worked examples.

Example 4.1

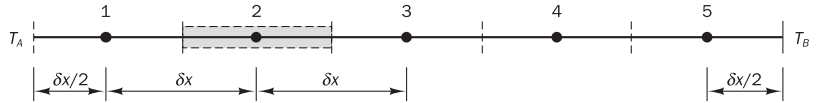
Consider the problem of source-free heat conduction in an insulated rod whose ends are maintained at constant temperatures of 100°C and 500°C respectively. The one-dimensional problem sketched in Figure 4.3 is governed by

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right) = 0 \tag{4.13}$$

Calculate the steady state temperature distribution in the rod. Thermal conductivity k equals 1000 W/m.K, cross-sectional area A is 10×10^{-3} m^2.

Figure 4.3

Solution

Let us divide the length of the rod into five equal control volumes as shown in Figure 4.4. This gives $\delta x = 0.1$ m.

Figure 4.4 The grid used

The grid consists of five nodes. For each one of nodes 2, 3 and 4 temperature values to the east and west are available as nodal values. Consequently, discretised equations of the form (4.10) can be readily written for control volumes surrounding these nodes:

$$\left(\frac{k_e}{\delta x_{PE}}A_e + \frac{k_w}{\delta x_{WP}}A_w\right)T_P = \left(\frac{k_w}{\delta x_{WP}}A_w\right)T_W + \left(\frac{k_e}{\delta x_{PE}}A_e\right)T_E \tag{4.14}$$

The thermal conductivity ($k_e = k_w = k$), node spacing (δx) and cross-sectional area ($A_e = A_w = A$) are constants. Therefore the **discretised equation for nodal points 2, 3 and 4** is

$$\boxed{a_P T_P = a_W T_W + a_E T_E} \tag{4.15}$$

with

a_W	a_E	a_P
$\frac{k}{\delta x}A$	$\frac{k}{\delta x}A$	$a_W + a_E$

S_u and S_p are zero in this case since there is no source term in the governing equation (4.13).

Nodes 1 and 5 are boundary nodes, and therefore require special attention. Integration of equation (4.13) over the control volume surrounding point 1 gives

$$kA\left(\frac{T_E - T_P}{\delta x}\right) - kA\left(\frac{T_P - T_A}{\delta x/2}\right) = 0 \tag{4.16}$$

This expression shows that the flux through control volume boundary A has been approximated by assuming a linear relationship between temperatures at boundary point A and node P. We can rearrange (4.16) as follows:

$$\left(\frac{k}{\delta x}A + \frac{2k}{\delta x}A\right)T_P = 0\ .\ T_W + \left(\frac{k}{\delta x}A\right)T_E + \left(\frac{2k}{\delta x}A\right)T_A \tag{4.17}$$

Comparing equation (4.17) with equation (4.10), it can be easily identified that the fixed temperature boundary condition enters the calculation as a source term $(S_u + S_P T_P)$ with $S_u = (2kA/\delta x)T_A$ and $S_p = -2kA/\delta x$, and that the link to the (west) boundary side has been suppressed by setting coefficient a_W to zero.

Equation (4.17) may be cast in the same form as (4.11) to yield the **discretised equation for boundary node 1**:

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.18}$$

with

a_W	a_E	a_P	S_P	S_u
0	$\frac{kA}{\delta x}$	$a_W + a_E - S_p$	$-\frac{2kA}{\delta x}$	$\frac{2kA}{\delta x}T_A$

The control volume surrounding node 5 can be treated in a similar manner. Its discretised equation is given by

$$kA\left(\frac{T_B - T_P}{\delta x/2}\right) - kA\left(\frac{T_P - T_W}{\delta x}\right) = 0 \tag{4.19}$$

As before we assume a linear temperature distribution between node P and boundary point B to approximate the heat flux through the control volume boundary. Equation (4.19) can be rearranged as

$$\left(\frac{k}{\delta x}A + \frac{2k}{\delta x}A\right)T_P = \left(\frac{k}{\delta x}A\right)T_W + 0\ .\ T_E + \left(\frac{2k}{\delta x}A\right)T_B \tag{4.20}$$

The **discretised equation for boundary node 5** is

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.21}$$

where

a_W	a_E	a_P	S_P	S_u
$\frac{kA}{\delta x}$	0	$a_W + a_E - S_p$	$-\frac{2kA}{\delta x}$	$\frac{2kA}{\delta x}T_B$

The discretisation process has yielded one equation for each of the nodal points 1 to 5. Substitution of numerical values gives $kA/\delta x = 100$, and the coefficients of each discretised equation can easily be worked out. Their values are given in Table 4.1.

The resulting set of algebraic equations for this example is

$$\begin{aligned}
300T_1 &= 100T_2 + 200T_A \\
200T_2 &= 100T_1 + 100T_3 \\
200T_3 &= 100T_2 + 100T_4 \\
200T_4 &= 100T_3 + 100T_5 \\
300T_5 &= 100T_4 + 200T_B
\end{aligned} \tag{4.22}$$

Table 4.1

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	100	$200T_A$	−200	300
2	100	100	0	0	200
3	100	100	0	0	200
4	100	100	0	0	200
5	100	0	$200T_B$	−200	300

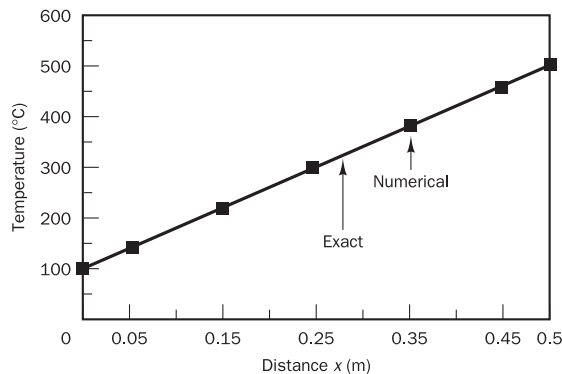
This set of equations can be rearranged as

$$\begin{bmatrix} 300 & -100 & 0 & 0 & 0 \\ -100 & 200 & -100 & 0 & 0 \\ 0 & -100 & 200 & -100 & 0 \\ 0 & 0 & -100 & 200 & -100 \\ 0 & 0 & 0 & -100 & 300 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 200T_A \\ 0 \\ 0 \\ 0 \\ 200T_B \end{bmatrix} \tag{4.23}$$

The above set of equations yields the steady state temperature distribution of the given situation. For simple problems involving a small number of nodes the resulting matrix equation can easily be solved with a software package such as MATLAB (1992). For $T_A = 100$ and $T_B = 500$ the solution of (4.23) can obtained by using, for example, Gaussian elimination:

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 140 \\ 220 \\ 300 \\ 380 \\ 460 \end{bmatrix} \tag{4.24}$$

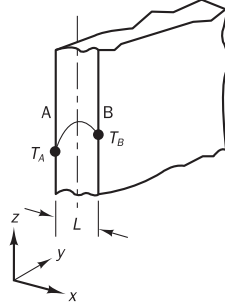
The exact solution is a linear distribution between the specified boundary temperatures: $T = 800x + 100$. Figure 4.5 shows that the exact solution and the numerical results coincide.

Figure 4.5 Comparison of the numerical result with the analytical solution

Example 4.2

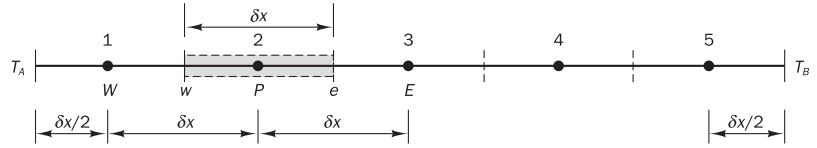
Now we discuss a problem that includes sources other than those arising from boundary conditions. Figure 4.6 shows a large plate of thickness $L = 2$ cm with constant thermal conductivity $k = 0.5$ W/m.K and uniform heat generation $q = 1000$ kW/m^3. The faces A and B are at temperatures of 100°C and 200°C respectively. Assuming that the dimensions in the y- and

z-directions are so large that temperature gradients are significant in the x-direction only, calculate the steady state temperature distribution. Compare the numerical result with the analytical solution. The governing equation is

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right)+q=0 \tag{4.25}$$

Figure 4.6

Solution

As before, the method of solution is demonstrated using a simple grid. The domain is divided into five control volumes (see Figure 4.7), giving $\delta x = 0.004$ m; a unit area is considered in the y–z plane.

Figure 4.7 The grid used

Formal integration of the governing equation over a control volume gives

$$\int_{\Delta V}\frac{\mathrm{d}}{\mathrm{d}x}\left(k\frac{\mathrm{d}T}{\mathrm{d}x}\right)\mathrm{d}V+\int_{\Delta V} q\,\mathrm{d}V=0 \tag{4.26}$$

We treat the first term of the above equation as in the previous example. The second integral, the source term of the equation, is evaluated by calculating the average generation (i.e. $\bar{S}\Delta V = q\Delta V$) within each control volume. Equation (4.26) can be written as

$$\left[\left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_e-\left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_w\right]+q\Delta V=0 \tag{4.27}$$

$$\left[k_eA\left(\frac{T_E-T_P}{\delta x}\right)-k_wA\left(\frac{T_P-T_W}{\delta x}\right)\right]+qA\delta x=0 \tag{4.28}$$

The above equation can be rearranged as

$$\left(\frac{k_eA}{\delta x}+\frac{k_wA}{\delta x}\right)T_P=\left(\frac{k_wA}{\delta x}\right)T_W+\left(\frac{k_eA}{\delta x}\right)T_E+qA\delta x \tag{4.29}$$

This equation is written in the general form of (4.11):

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.30}$$

Since $k_e = k_w = k$ we have the following coefficients:

a_W	a_E	a_P	S_P	S_u
$\frac{kA}{\delta x}$	$\frac{kA}{\delta x}$	$a_W + a_E - S_P$	0	$qA\delta x$

Equation (4.30) is valid for control volumes at **nodal points 2, 3 and 4**.

To incorporate the boundary conditions at nodes 1 and 5 we apply the linear approximation for temperatures between a boundary point and the adjacent nodal point. At node 1 the temperature at the west boundary is known. Integration of equation (4.25) at the control volume surrounding node 1 gives

$$\left[\left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_e - \left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_w\right] + q\Delta V = 0 \tag{4.31}$$

Introduction of the linear approximation for temperatures between A and P yields

$$\left[k_e A\left(\frac{T_E - T_P}{\delta x}\right) - k_A A\left(\frac{T_P - T_A}{\delta x/2}\right)\right] + qA\delta x = 0 \tag{4.32}$$

The above equation can be rearranged, using $k_e = k_A = k$, to yield the discretised equation for **boundary node 1**:

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.33}$$

where

a_W	a_E	a_P	S_P	S_u
0	$\frac{kA}{\delta x}$	$a_W + a_E - S_P$	$-\frac{2kA}{\delta x}$	$qA\delta x + \frac{2kA}{\delta x}T_A$

At nodal point 5, the temperature on the east face of the control volume is known. The node is treated in a similar way to boundary node 1. At boundary point 5 we have

$$\left[\left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_e - \left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right)_w\right] + q\Delta V = 0 \tag{4.34}$$

$$\left[k_B A\left(\frac{T_B - T_P}{\delta x/2}\right) - k_w A\left(\frac{T_P - T_W}{\delta x}\right)\right] + qA\delta x = 0 \tag{4.35}$$

The above equation can be rearranged, noting that $k_B = k_w = k$, to give the discretised equation for **boundary node 5**:

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.36}$$

where

a_W	a_E	a_P	S_P	S_u
$\frac{kA}{\delta x}$	0	$a_W + a_E - S_P$	$-\frac{2kA}{\delta x}$	$qA\delta x + \frac{2kA}{\delta x}T_B$

Substitution of numerical values for $A = 1$, $k = 0.5$ W/m.K, $q = 1000$ kW/m^3 and $\delta x = 0.004$ m everywhere gives the coefficients of the discretised equations summarised in Table 4.2.

Table 4.2

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	125	$4000 + 250T_A$	−250	375
2	125	125	4000	0	250
3	125	125	4000	0	250
4	125	125	4000	0	250
5	125	0	$4000 + 250T_B$	−250	375

Given directly in matrix form the equations are

$$\begin{bmatrix} 375 & -125 & 0 & 0 & 0 \\ -125 & 250 & -125 & 0 & 0 \\ 0 & -125 & 250 & -125 & 0 \\ 0 & 0 & -125 & 250 & -125 \\ 0 & 0 & 0 & -125 & 375 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 29000 \\ 4000 \\ 4000 \\ 4000 \\ 54000 \end{bmatrix} \tag{4.37}$$

The solution to the above set of equations is

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 150 \\ 218 \\ 254 \\ 258 \\ 230 \end{bmatrix} \tag{4.38}$$

Comparison with the analytical solution

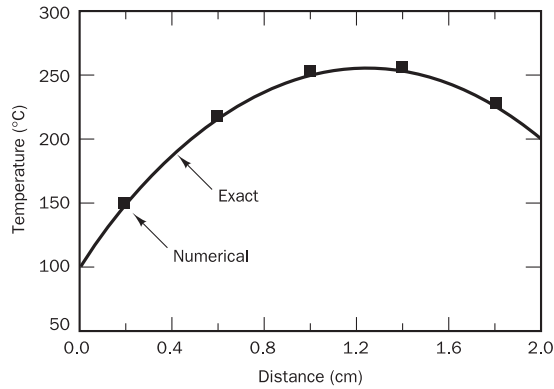
The analytical solution to this problem may be obtained by integrating equation (4.25) twice with respect to x and by subsequent application of the boundary conditions. This gives

$$T = \left[\frac{T_B - T_A}{L} + \frac{q}{2k}(L - x) \right] x + T_A \tag{4.39}$$

The comparison between the finite volume solution and the exact solution is shown in Table 4.3 and Figure 4.8 and it can be seen that, even with a coarse grid of five nodes, the agreement is very good.

Table 4.3

Node number	*1*	*2*	*3*	*4*	*5*
x (m)	0.002	0.006	0.01	0.014	0.018
Finite volume solution	150	218	254	258	230
Exact solution	146	214	250	254	226
Percentage error	2.73	1.86	1.60	1.57	1.76

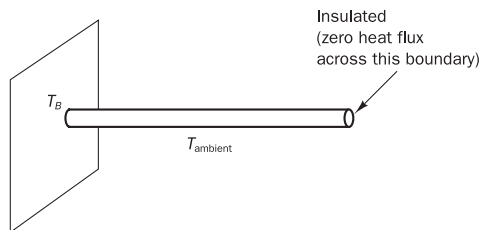
Figure 4.8 Comparison of the numerical result with the analytical solution

Example 4.3

In the final worked example of this chapter we discuss the cooling of a circular fin by means of convective heat transfer along its length. Convection gives rise to a temperature-dependent heat loss or sink term in the governing equation. Shown in Figure 4.9 is a cylindrical fin with uniform cross-sectional area A. The base is at a temperature of 100°C (T_B) and the end is insulated. The fin is exposed to an ambient temperature of 20°C. One-dimensional heat transfer in this situation is governed by

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(kA\frac{\mathrm{d}T}{\mathrm{d}x}\right) - hP(T - T_\infty) = 0 \tag{4.40}$$

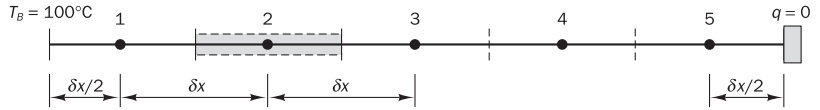
where h is the convective heat transfer coefficient, P the perimeter, k the thermal conductivity of the material and T_∞ the ambient temperature. Calculate the temperature distribution along the fin and compare the results with the analytical solution given by

$$\frac{T - T_\infty}{T_B - T_\infty} = \frac{\cosh[n(L - x)]}{\cosh(nL)} \tag{4.41}$$

Figure 4.9 The geometry for Example 4.3

where $n^2 = hP/(kA)$, L is the length of the fin and x the distance along the fin. Data: $L = 1$ m, $hP/(kA) = 25/\text{m}^2$ (note that kA is constant).

Solution

The governing equation in the example contains a sink term, $-hP(T - T_\infty)$, the convective heat loss, which is a function of the local temperature T. As before, the first step in solving the problem by the finite volume method is to set up a grid. We use a uniform grid and divide the length into five control volumes so that $\delta x = 0.2$ m. The grid is shown in Figure 4.10.

Figure 4.10 The grid used in Example 4.3

When kA = constant, the governing equation (4.40) can be written as

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\mathrm{d}T}{\mathrm{d}x}\right) - n^2(T - T_\infty) = 0 \text{ where } n^2 = hp/(kA) \tag{4.42}$$

Integration of the above equation over a control volume gives

$$\int_{\Delta V} \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\mathrm{d}T}{\mathrm{d}x}\right)\mathrm{d}V - \int_{\Delta V} n^2(T - T_\infty)\mathrm{d}V = 0 \tag{4.43}$$

The first integral of the above equation is treated as in Examples 4.1 and 4.2; the second integral due to the source term in the equation is evaluated by assuming that the integrand is locally constant within each control volume:

$$\left[\left(A\frac{\mathrm{d}T}{\mathrm{d}x}\right)_e - \left(A\frac{\mathrm{d}T}{\mathrm{d}x}\right)_w\right] - [n^2(T_P - T_\infty)A\delta x] = 0$$

First we develop a formula valid for nodal points 2, 3 and 4 by introducing the usual linear approximations for the temperature gradient. Subsequent division by cross-sectional area A gives

$$\left[\left(\frac{T_E - T_P}{\delta x}\right) - \left(\frac{T_P - T_W}{\delta x}\right)\right] - [n^2(T_P - T_\infty)\delta x] = 0 \tag{4.44}$$

This can be rearranged as

$$\left(\frac{1}{\delta x} + \frac{1}{\delta x}\right)T_P = \left(\frac{1}{\delta x}\right)T_W + \left(\frac{1}{\delta x}\right)T_E + n^2\delta x T_\infty - n^2\delta x T_P \tag{4.45}$$

For **interior nodal points 2, 3 and 4** we write, using general form (4.11),

$$\boxed{a_P T_P = a_W T_W + a_E T_E + S_u} \tag{4.46}$$

with

a_W	a_E	a_P	S_P	S_u
$\frac{1}{\delta x}$	$\frac{1}{\delta x}$	$a_W + a_E - S_P$	$-n^2\delta x$	$n^2\delta x T_\infty$

Next we apply the boundary conditions at nodal points 1 and 5. At node 1 the west control volume boundary is kept at a specified temperature. It is treated in the same way as in Example 4.1, i.e.

$$\left[\left(\frac{T_E - T_P}{\delta x}\right) - \left(\frac{T_P - T_B}{\delta x/2}\right)\right] - [n^2(T_P - T_\infty)\delta x] = 0 \tag{4.47}$$

The coefficients of the discretised equation at **boundary node 1** are

a_W	a_E	a_P	S_P	S_u
0	$\frac{1}{\delta x}$	$a_W + a_E - S_P$	$-n^2\delta x - \frac{2}{\delta x}$	$n^2\delta x T_\infty + \frac{2}{\delta x}T_B$

At node 5 the flux across the east boundary is zero since the east side of the control volume is an insulated boundary:

$$\left[0 - \left(\frac{T_P - T_W}{\delta x}\right)\right] - [n^2(T_P - T_\infty)\delta x] = 0 \tag{4.48}$$

Hence the east coefficient is set to zero. There are no additional source terms associated with the zero flux boundary condition. The coefficients at **boundary node 5** are given by

a_W	a_E	a_P	S_P	S_u
$\frac{1}{\delta x}$	0	$a_W + a_E - S_P$	$-n^2\delta x$	$n^2\delta x T_\infty$

Substituting numerical values gives the coefficients in Table 4.4.

Table 4.4

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	5	$100 + 10T_B$	−15	20
2	5	5	100	−5	15
3	5	5	100	−5	15
4	5	5	100	−5	15
5	5	0	100	−5	10

The matrix form of the equations set is

$$\begin{bmatrix} 20 & -5 & 0 & 0 & 0 \\ -5 & 15 & -5 & 0 & 0 \\ 0 & -5 & 15 & -5 & 0 \\ 0 & 0 & -5 & 15 & -5 \\ 0 & 0 & 0 & -5 & 10 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 1100 \\ 100 \\ 100 \\ 100 \\ 100 \end{bmatrix} \tag{4.49}$$

The solution to the above system is

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 64.22 \\ 36.91 \\ 26.50 \\ 22.60 \\ 21.30 \end{bmatrix} \tag{4.50}$$

Comparison with the analytical solution

Table 4.5 compares the finite volume solution with analytical expression (4.41). The maximum percentage error ((analytical solution – finite volume solution)/analytical solution) is around 6%. Given the coarseness of the grid used in the calculation, the numerical solution is reasonably close to the exact solution.

Table 4.5

Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.1	64.22	68.52	4.30	6.27
2	0.3	36.91	37.86	0.95	2.51
3	0.5	26.50	26.61	0.11	0.41
4	0.7	22.60	22.53	−0.07	−0.31
5	0.9	21.30	21.21	−0.09	−0.42

The numerical solution can be improved by employing a finer grid. Let us consider the same problem with the rod length sub-divided into 10 control volumes. The derivation of the discretised equations is the same as before, but the numerical values of the coefficients and source terms are different due to the smaller grid spacing of $\delta x = 0.1$ m. The comparison of results of the second calculation with the analytical solution is shown in Figure 4.11 and Table 4.6. The second numerical results shows better agreement with the analytical solution; now the maximum deviation is only 2%.

Figure 4.11 Comparison of numerical and analytical results

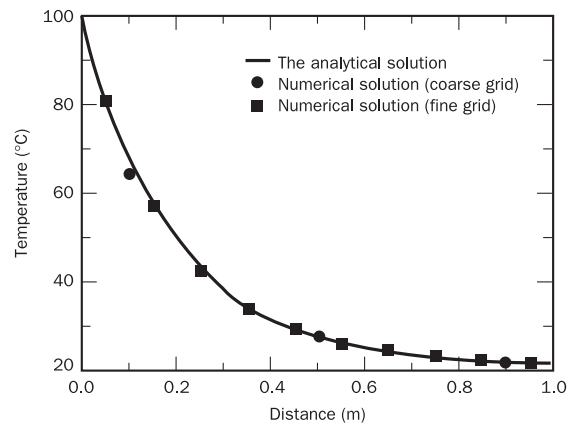



Table 4.6

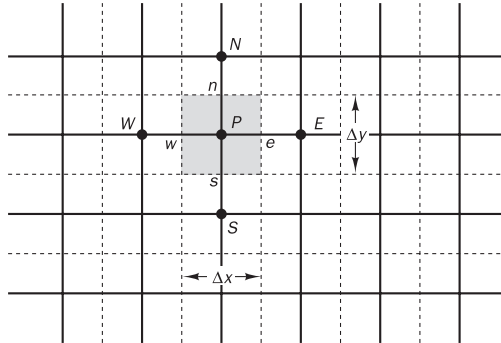
Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.05	80.59	82.31	1.72	2.08
2	0.15	56.94	57.79	0.85	1.47
3	0.25	42.53	42.93	0.40	0.93
4	0.35	33.74	33.92	0.18	0.53
5	0.45	28.40	28.46	0.06	0.21
6	0.55	25.16	25.17	0.01	0.03
7	0.65	23.21	23.19	−0.02	−0.08
8	0.75	22.06	22.03	−0.03	−0.13
9	0.85	21.47	21.39	−0.08	−0.37
10	0.95	21.13	21.11	−0.02	−0.09

4.4 Finite volume method for two-dimensional diffusion problems

The methodology used in deriving discretised equations in the one-dimensional case can be easily extended to two-dimensional problems. To illustrate the technique let us consider the two-dimensional steady state diffusion equation given by

$$\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial\phi}{\partial y}\right)+S_\phi=0 \tag{4.51}$$

A portion of the two-dimensional grid used for the discretisation is shown in Figure 4.12.

Figure 4.12 A part of the two-dimensional grid

In addition to the east (E) and west (W) neighbours a general grid node P now also has north (N) and south (S) neighbours. The same notation as in the one-dimensional analysis is used for faces and cell dimensions. When the above equation is formally integrated over the control volume we obtain

$$\int_{\Delta V}\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)\mathrm{d}x\,.\,\mathrm{d}y+\int_{\Delta V}\frac{\partial}{\partial y}\left(\Gamma\frac{\partial\phi}{\partial y}\right)\mathrm{d}x\,.\,\mathrm{d}y+\int_{\Delta V}S_\phi\mathrm{d}V=0 \tag{4.52}$$

So, noting that $A_e = A_w = \Delta y$ and $A_n = A_s = \Delta x$, we obtain

$$\left[\Gamma_e A_e\left(\frac{\partial\phi}{\partial x}\right)_e - \Gamma_w A_w\left(\frac{\partial\phi}{\partial x}\right)_w\right] + \left[\Gamma_n A_n\left(\frac{\partial\phi}{\partial y}\right)_n - \Gamma_s A_s\left(\frac{\partial\phi}{\partial y}\right)_s\right] + \bar{S}\Delta V = 0 \tag{4.53}$$

As before, this equation represents the balance of the generation of ϕ in a control volume and the fluxes through its cell faces. Using the approximations introduced in the previous section we can write expressions for the flux through control volume faces:

$$\text{Flux across the west face} = \Gamma_w A_w \left.\frac{\partial\phi}{\partial x}\right|_w = \Gamma_w A_w \frac{(\phi_P - \phi_W)}{\delta x_{WP}} \tag{4.54a}$$

$$\text{Flux across the east face} = \Gamma_e A_e \left.\frac{\partial\phi}{\partial x}\right|_e = \Gamma_e A_e \frac{(\phi_E - \phi_P)}{\delta x_{PE}} \tag{4.54b}$$

$$\text{Flux across the south face} = \Gamma_s A_s \left.\frac{\partial\phi}{\partial y}\right|_s = \Gamma_s A_s \frac{(\phi_P - \phi_S)}{\delta y_{SP}} \tag{4.54c}$$

$$\text{Flux across the north face} = \Gamma_n A_n \left.\frac{\partial\phi}{\partial y}\right|_n = \Gamma_n A_n \frac{(\phi_N - \phi_P)}{\delta y_{PN}} \tag{4.54d}$$

By substituting the above expressions into equation (4.53) we obtain

$$\Gamma_e A_e \frac{(\phi_E - \phi_P)}{\delta x_{PE}} - \Gamma_w A_w \frac{(\phi_P - \phi_W)}{\delta x_{WP}} + \Gamma_n A_n \frac{(\phi_N - \phi_P)}{\delta y_{PN}} - \Gamma_s A_s \frac{(\phi_P - \phi_S)}{\delta y_{SP}} + \bar{S}\Delta V = 0 \tag{4.55}$$

When the source term is represented in the linearised form $\bar{S}\Delta V = S_u + S_p\phi_P$, this equation can be rearranged as

$$\left(\frac{\Gamma_w A_w}{\delta x_{WP}} + \frac{\Gamma_e A_e}{\delta x_{PE}} + \frac{\Gamma_s A_s}{\delta y_{SP}} + \frac{\Gamma_n A_n}{\delta y_{PN}} - S_p\right)\phi_P = \left(\frac{\Gamma_w A_w}{\delta x_{WP}}\right)\phi_W + \left(\frac{\Gamma_e A_e}{\delta x_{PE}}\right)\phi_E + \left(\frac{\Gamma_s A_s}{\delta y_{SP}}\right)\phi_S + \left(\frac{\Gamma_n A_n}{\delta y_{PN}}\right)\phi_N + S_u \tag{4.56}$$

Equation (4.56) is now cast in the general discretised equation form for interior nodes:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + S_u} \tag{4.57}$$

where

a_W	a_E	a_S	a_N	a_P
$\dfrac{\Gamma_w A_w}{\delta x_{WP}}$	$\dfrac{\Gamma_e A_e}{\delta x_{PE}}$	$\dfrac{\Gamma_s A_s}{\delta y_{SP}}$	$\dfrac{\Gamma_n A_n}{\delta y_{PN}}$	$a_W + a_E + a_S + a_N - S_p$

The face areas in a two-dimensional case are $A_w = A_e = \Delta y$; $A_n = A_s = \Delta x$.

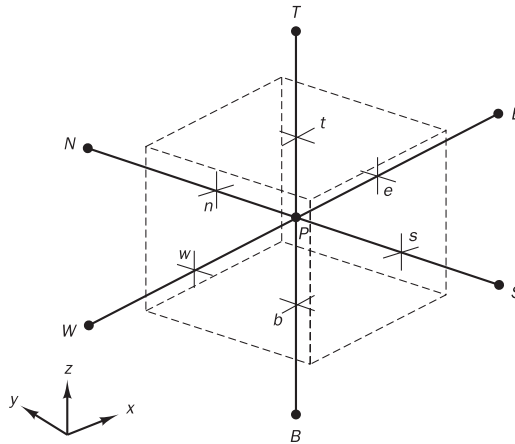
We obtain the distribution of the property ϕ in a given two-dimensional situation by writing discretised equations of the form (4.57) at each grid node of the sub-divided domain. At the boundaries where the temperatures or fluxes are known the discretised equations are modified to incorporate boundary conditions in the manner demonstrated in Examples 4.1 and 4.2. The boundary-side coefficient is set to zero (cutting the link with the boundary) and the flux crossing the boundary is introduced as a source which is appended to any existing S_u and S_p terms. Subsequently, the resulting set of equations is solved to obtain the two-dimensional distribution of the property ϕ. Example 7.2 in Chapter 7 shows how the method can be applied to calculate conductive heat transfer in two-dimensional situations.

4.5 Finite volume method for three-dimensional diffusion problems

Steady state diffusion in a three-dimensional situation is governed by

$$\frac{\partial}{\partial x}\left(\Gamma\frac{\partial \phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma\frac{\partial \phi}{\partial y}\right) + \frac{\partial}{\partial z}\left(\Gamma\frac{\partial \phi}{\partial z}\right) + S_\phi = 0 \tag{4.58}$$

Now a three-dimensional grid is used to sub-divide the domain. A typical control volume is shown in Figure 4.13.

Figure 4.13 A cell in three dimensions and neighbouring nodes

A cell containing node P now has six neighbouring nodes identified as west, east, south, north, bottom and top (W, E, S, N, B, T). As before, the notation w, e, s, n, b and t is used to refer to the west, east, south, north, bottom and top cell faces respectively.

Integration of Equation (4.58) over the control volume shown gives

$$\left[\Gamma_e A_e\left(\frac{\partial\phi}{\partial x}\right)_e - \Gamma_w A_w\left(\frac{\partial\phi}{\partial x}\right)_w\right] + \left[\Gamma_n A_n\left(\frac{\partial\phi}{\partial y}\right)_n - \Gamma_s A_s\left(\frac{\partial\phi}{\partial y}\right)_s\right] + \left[\Gamma_t A_t\left(\frac{\partial\phi}{\partial z}\right)_t - \Gamma_b A_b\left(\frac{\partial\phi}{\partial z}\right)_b\right] + \bar{S}\Delta V = 0 \tag{4.59}$$

Following the procedure developed for one- and two-dimensional cases the discretised form of equation (4.59) is obtained:

$$\left[\Gamma_e A_e\frac{(\phi_E-\phi_P)}{\delta x_{PE}} - \Gamma_w A_w\frac{(\phi_P-\phi_W)}{\delta x_{WP}}\right] + \left[\Gamma_n A_n\frac{(\phi_N-\phi_P)}{\delta y_{PN}} - \Gamma_s A_s\frac{(\phi_P-\phi_S)}{\delta y_{SP}}\right] + \left[\Gamma_t A_t\frac{(\phi_T-\phi_P)}{\delta z_{PT}} - \Gamma_b A_b\frac{(\phi_P-\phi_B)}{\delta z_{BP}}\right] + (S_u + S_P\phi_P) = 0 \tag{4.60}$$

As before, this can be rearranged to give the discretised equation for interior nodes:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + a_B\phi_B + a_T\phi_T + S_u} \tag{4.61}$$

where

a_W	a_E	a_S	a_N	a_B	a_T	a_P
$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	$\frac{\Gamma_b A_b}{\delta z_{BP}}$	$\frac{\Gamma_t A_t}{\delta z_{PT}}$	$a_W + a_E + a_S + a_N + a_B + a_T - S_P$

Boundary conditions can be introduced by cutting links with the appropriate face(s) and modifying the source term as described in section 4.3.

4.6 Summary

- The discretised equations for one-, two- and three-dimensional diffusion problems have been found to take the following general form:

$$\boxed{a_P\phi_P = \sum a_{nb}\phi_{nb} + S_u} \tag{4.62}$$

where Σ indicates summation over all neighbouring nodes (nb), a_{nb} are the neighbouring coefficients, a_W, a_E in one dimension, a_W, a_E, a_S, a_N in two dimensions and a_W, a_E, a_S, a_N, a_B, a_T in three dimensions; ϕ_{nb} are the values of the property ϕ at the neighbouring nodes; and $(S_u + S_P\phi_P)$ is the linearised source term.

- In all cases the coefficients around point P satisfy the following relation:

$$\boxed{a_P = \sum a_{nb} - S_p} \tag{4.63}$$

Table 4.7

	a_W	a_E	a_S	a_N	a_B	a_T
1D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	–	–	–	–
2D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	–	–
3D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	$\frac{\Gamma_b A_b}{\delta z_{BP}}$	$\frac{\Gamma_t A_t}{\delta z_{PT}}$

- A summary of the neighbour coefficients for one-, two- and three-dimensional diffusion problems is given in Table 4.7.
- Source terms can be included by identifying their linearised form $\bar{S}\Delta V = S_u + S_p\phi_P$ and specifying values for S_u and S_p.
- Boundary conditions are incorporated by suppressing the link to the boundary side and introducing the boundary side flux – exact or linearly approximated – through additional source terms S_u and S_p. For a one-dimensional control volume of width $\Delta\varsigma$ with a boundary B:
  - link cutting:

$$\text{set coefficient } a_B = 0 \tag{4.64}$$

  - source contributions:

$$\text{fixed value } \phi_B\text{: } S_u = \frac{2k_B A_B}{\Delta\zeta}\phi_B$$

$$S_p = -\frac{2k_B A_B}{\Delta\zeta} \tag{4.65}$$

$$\text{fixed flux } q_B\text{: } \quad S_u + S_p\phi_P = q_B \tag{4.66}$$

Chapter five The finite volume method for convection–diffusion problems

5.1 Introduction

In problems where fluid flow plays a significant role we must account for the effects of convection. Diffusion always occurs alongside convection in nature so here we examine methods to predict combined convection and diffusion. The steady convection–diffusion equation can be derived from the transport equation (2.39) for a general property ϕ by deleting the transient term

$$\operatorname{div}(\rho \mathbf{u} \phi) = \operatorname{div}(\Gamma \operatorname{grad} \phi) + S_\phi \tag{5.1}$$

Formal integration over a control volume gives

$$\int_A \mathbf{n} \,.\, (\rho \phi \mathbf{u}) \mathrm{d}A = \int_A \mathbf{n} \,.\, (\Gamma \operatorname{grad} \phi) \mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \tag{5.2}$$

This equation represents the flux balance in a control volume. The left hand side gives the net convective flux and the right hand side contains the net diffusive flux and the generation or destruction of the property ϕ within the control volume.

The principal problem in the discretisation of the convective terms is the calculation of the value of transported property ϕ at control volume faces and its convective flux across these boundaries. In Chapter 4 we introduced the central differencing method of obtaining discretised equations for the diffusion and source terms on the right hand side of equation (5.2). It would seem obvious to try out this practice, which worked so well for diffusion problems, on the convective terms. However, the diffusion process affects the distribution of a transported quantity along its gradients in all directions, whereas convection spreads influence only in the flow direction. This crucial difference manifests itself in a stringent upper limit to the grid size, which is dependent on the relative strength of convection and diffusion, for stable convection–diffusion calculations with central differencing.

Naturally, we also present the case for a number of alternative discretisation practices for the convective effects which enable stable computations under less restrictive conditions. In the current analysis no reference will be made to the evaluation of face velocities. It is assumed that they are 'somehow' known. The method of computing velocities will be discussed in Chapter 6.

5.2 Steady one-dimensional convection and diffusion

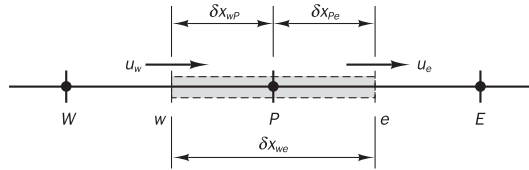
In the absence of sources, steady convection and diffusion of a property ϕ in a given one-dimensional flow field u is governed by

$$\frac{d}{dx}(\rho u\phi) = \frac{d}{dx}\left(\Gamma\frac{d\phi}{dx}\right) \tag{5.3}$$

The flow must also satisfy continuity, so

$$\frac{d(\rho u)}{dx} = 0 \tag{5.4}$$

We consider the one-dimensional control volume shown in Figure 5.1 and use the notation introduced in Chapter 4. Our attention is focused on a general node P; the neighbouring nodes are identified by W and E and the control volume faces by w and e.

Figure 5.1 A control volume around node P

Integration of transport equation (5.3) over the control volume of Figure 5.1 gives

$$(\rho uA\phi)_e - (\rho uA\phi)_w = \left(\Gamma A\frac{d\phi}{dx}\right)_e - \left(\Gamma A\frac{d\phi}{dx}\right)_w \tag{5.5}$$

And integration of continuity equation (5.4) yields

$$(\rho uA)_e - (\rho uA)_w = 0 \tag{5.6}$$

To obtain discretised equations for the convection–diffusion problem we must approximate the terms in equation (5.5). It is convenient to define two variables F and D to represent the convective mass flux per unit area and diffusion conductance at cell faces:

$$F = \rho u \text{ and } D = \frac{\Gamma}{\delta x} \tag{5.7}$$

The cell face values of the variables F and D can be written as

$$F_w = (\rho u)_w \qquad F_e = (\rho u)_e \tag{5.8a}$$

$$D_w = \frac{\Gamma_w}{\delta x_{WP}} \qquad D_e = \frac{\Gamma_e}{\delta x_{PE}} \tag{5.8b}$$

We develop our techniques assuming that $A_w = A_e = A$, so we can divide the left and right hand sides of equation (5.5) by area A. As before, we employ the central differencing approach to represent the contribution of the diffusion terms on the right hand side. The integrated convection–diffusion equation (5.5) can now be written as

$$\boxed{F_e\phi_e - F_w\phi_w = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)} \tag{5.9}$$

and the integrated continuity equation (5.6) as

$$\boxed{F_e - F_w = 0} \tag{5.10}$$

We also assume that the velocity field is 'somehow known', which takes care of the values of F_e and F_w. In order to solve equation (5.9) we need to calculate the transported property ϕ at the e and w faces. Schemes for this purpose are assessed in the following sections.

5.3 The central differencing scheme

The central differencing approximation has been used to represent the diffusion terms which appear on the right hand side of equation (5.9), and it seems logical to try linear interpolation to compute the cell face values for the convective terms on the left hand side of this equation. For a uniform grid we can write the cell face values of property ϕ as

$$\phi_e = (\phi_P + \phi_E)/2 \tag{5.11a}$$

$$\phi_w = (\phi_W + \phi_P)/2 \tag{5.11b}$$

Substitution of the above expressions into the convection terms of (5.9) yields

$$\frac{F_e}{2}(\phi_P + \phi_E) - \frac{F_w}{2}(\phi_W + \phi_P) = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.12}$$

This can be rearranged to give

$$\left[\left(D_w - \frac{F_w}{2}\right) + \left(D_e + \frac{F_e}{2}\right)\right]\phi_P = \left(D_w + \frac{F_w}{2}\right)\phi_W + \left(D_e - \frac{F_e}{2}\right)\phi_E$$

$$\left[\left(D_w + \frac{F_w}{2}\right) + \left(D_e - \frac{F_e}{2}\right) + (F_e - F_w)\right]\phi_P$$

$$= \left(D_w + \frac{F_w}{2}\right)\phi_W + \left(D_e - \frac{F_e}{2}\right)\phi_E \tag{5.13}$$

Identifying the coefficients of ϕ_W and ϕ_E as a_W and a_E, the **central differencing** expressions for the discretised convection–diffusion equation are

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E} \tag{5.14}$$

where

a_W	a_E	a_P
$D_w + \frac{F_w}{2}$	$D_e - \frac{F_e}{2}$	$a_W + a_E + (F_e - F_w)$

It can be easily recognised that equation (5.14) for steady convection–diffusion problems takes the same general form as equation (4.11) for pure diffusion problems. The difference is that the coefficients of the former contain

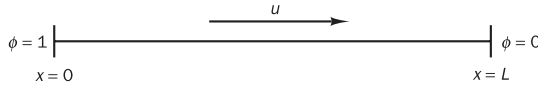
additional terms to account for convection. To solve a one-dimensional convection–diffusion problem we write discretised equations of the form (5.14) for all grid nodes. This yields a set of algebraic equations that is solved to obtain the distribution of the transported property ϕ. The process is now illustrated by means of a worked example.

Example 5.1

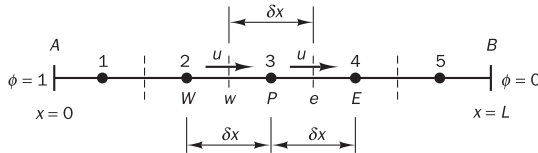
A property ϕ is transported by means of convection and diffusion through the one-dimensional domain sketched in Figure 5.2. The governing equation is (5.3); the boundary conditions are $\phi_0 = 1$ at $x = 0$ and $\phi_L = 0$ at $x = L$. Using five equally spaced cells and the central differencing scheme for convection and diffusion, calculate the distribution of ϕ as a function of x for (i) Case 1: u = 0.1 m/s, (ii) Case 2: u = 2.5 m/s, and compare the results with the analytical solution

$$\frac{\phi - \phi_0}{\phi_L - \phi_0} = \frac{\exp(\rho u x/\Gamma) - 1}{\exp(\rho u L/\Gamma) - 1} \tag{5.15}$$

(iii) Case 3: recalculate the solution for $u = 2.5$ m/s with 20 grid nodes and compare the results with the analytical solution. The following data apply: length $L = 1.0$ m, $\rho = 1.0$ kg/m^3, $\Gamma = 0.1$ kg/m.s.

Figure 5.2

Solution

The method of solution is demonstrated using the simple grid shown in Figure 5.3. The domain has been divided into five control volumes giving $\delta x = 0.2$ m. Note that $F = \rho u$, $D = \Gamma/\delta x$, $F_e = F_w = F$ and $D_e = D_w = D$ everywhere. The boundaries are denoted by A and B.

Figure 5.3 The grid used for discretisation

The discretisation equation (5.14) and its coefficients apply at internal nodal points 2, 3 and 4, but control volumes 1 and 5 need special treatment since they are adjacent to the domain boundaries. We integrate governing equation (5.3) and use central differencing for both the diffusion terms and the convective flux through the east face of cell 1. The value of ϕ is given at the west face of this cell ($\phi_w = \phi_A = 1$) so we do not need to make any approximations in the convective flux term at this boundary. This yields the following equation for node 1:

$$\frac{F_e}{2}(\phi_P + \phi_E) - F_A\phi_A = D_e(\phi_E - \phi_P) - D_A(\phi_P - \phi_A) \tag{5.16}$$

For control volume 5, the ϕ-value at the east face is known ($\phi_e = \phi_B = 0$). We obtain

$$F_B\phi_B - \frac{F_w}{2}(\phi_P + \phi_W) = D_B(\phi_B - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.17}$$

Rearrangement of equations (5.16) and (5.17), noting that $D_A = D_B = 2\Gamma/\delta x = 2D$ and $F_A = F_B = F$, gives discretised equations at boundary nodes of the following form:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u} \tag{5.18}$$

with central coefficient

$$\boxed{a_P = a_W + a_E + (F_e - F_w) - S_P}$$

and

Node	a_W	a_E	S_P	S_u
1	0	$D - F/2$	$-(2D + F)$	$(2D + F)\phi_A$
2, 3, 4	$D + F/2$	$D - F/2$	0	0
5	$D + F/2$	0	$-(2D - F)$	$(2D - F)\phi_B$

To introduce the boundary conditions we have suppressed the link to the boundary side and entered the boundary flux through the source terms.

(i) Case 1
$u = 0.1$ m/s: $F = \rho u = 0.1$, $D = \Gamma/\delta x = 0.1/0.2 = 0.5$ gives the coefficients as summarised in Table 5.1.

Table 5.1

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	0.45	$1.1\phi_A$	–1.1	1.55
2	0.55	0.45	0	0	1.0
3	0.55	0.45	0	0	1.0
4	0.55	0.45	0	0	1.0
5	0.55	0	$0.9\phi_B$	–0.9	1.45

The matrix form of the equation set using $\phi_A = 1$ and $\phi_B = 0$ is

$$\begin{bmatrix} 1.55 & -0.45 & 0 & 0 & 0 \\ -0.55 & 1.0 & -0.45 & 0 & 0 \\ 0 & -0.55 & 1.0 & -0.45 & 0 \\ 0 & 0 & -0.55 & 1.0 & -0.45 \\ 0 & 0 & 0 & -0.55 & 1.45 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 1.1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.19}$$

The solution to the above system is

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 0.9421 \\ 0.8006 \\ 0.6276 \\ 0.4163 \\ 0.1579 \end{bmatrix} \tag{5.20}$$

Comparison with the analytical solution

Substitution of the data into equation (5.15) gives the exact solution of the problem:

$$\phi(x) = \frac{2.7183 - \exp(x)}{1.7183}$$

The numerical and analytical solutions are compared in Table 5.2 and in Figure 5.4. Given the coarseness of the grid the central differencing (CD) scheme gives reasonable agreement with the analytical solution.

Table 5.2

Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.1	0.9421	0.9387	−0.003	−0.36
2	0.3	0.8006	0.7963	−0.004	−0.53
3	0.5	0.6276	0.6224	−0.005	−0.83
4	0.7	0.4163	0.4100	−0.006	−1.53
5	0.9	0.1579	0.1505	−0.007	−4.91

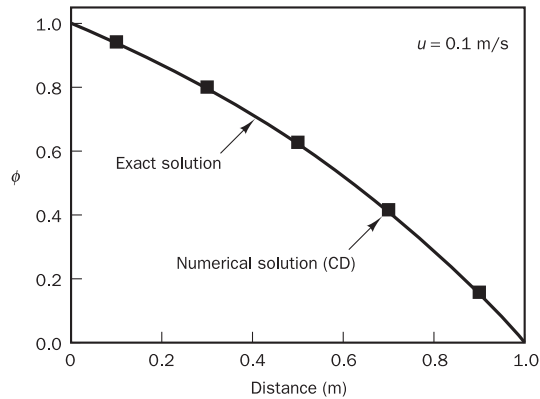
Figure 5.4 Comparison of numerical and analytical solutions for Case 1

(ii) Case 2
$u = 2.5$ m/s: $F = \rho u = 2.5$, $D = \Gamma/\delta x = 0.1/0.2 = 0.5$ gives the coefficients as summarised in Table 5.3.

Table 5.3

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	−0.75	$3.5\phi_A$	−3.5	2.75
2	1.75	−0.75	0	0	1.0
3	1.75	−0.75	0	0	1.0
4	1.75	−0.75	0	0	1.0
5	1.75	0	$-1.5\phi_B$	1.5	0.25

Comparison of numerical and analytical solution

The matrix equations are formed from the coefficients in Table 5.3 by the same method used in Case 1 and subsequently solved. The analytical solution for the data that apply here is

$$\phi(x) = 1 + \frac{1 - \exp(25x)}{7.20 \times 10^{10}}$$

The numerical and analytical solutions are compared in Table 5.4 and shown in Figure 5.5. The central differencing scheme produces a solution that appears to oscillate about the exact solution. These oscillations are often called 'wiggles' in the literature; the agreement with the analytical solution is clearly not very good.

Table 5.4

Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.1	1.0356	1.0000	−0.035	−3.56
2	0.3	0.8694	0.9999	0.131	13.05
3	0.5	1.2573	0.9999	−0.257	−25.74
4	0.7	0.3521	0.9994	0.647	64.70
5	0.9	2.4644	0.9179	−1.546	−168.48

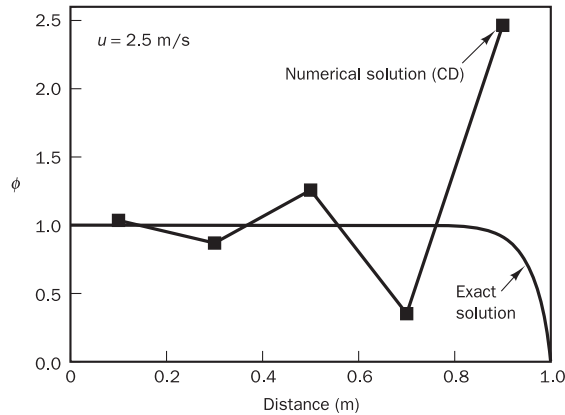
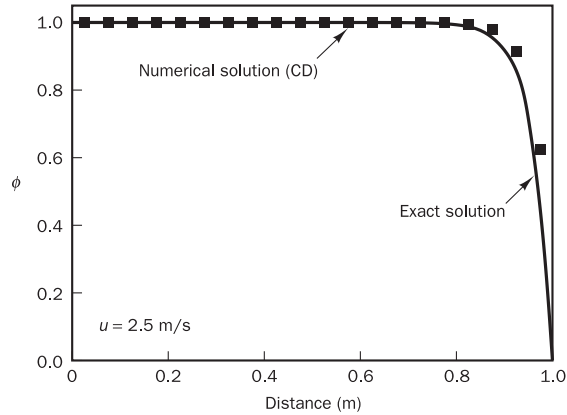
Figure 5.5 Comparison of numerical and analytical solutions for Case 2

(iii) Case 3
$u = 2.5$ m/s: a grid of 20 nodes gives $\delta x = 0.05$, $F = \rho u = 2.5$, $D = \Gamma/\delta x = 0.1/0.05 = 2.0$. The coefficients are summarised in Table 5.5 and the resulting solution is compared with the analytical solution in Figure 5.6.

Table 5.5

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	0.75	$6.5\phi_A$	−6.5	7.25
2–19	3.25	0.75	0	0	4.00
20	3.25	0	$1.5\phi_B$	−1.5	4.75

Figure 5.6 Comparison of numerical and analytical solutions for Case 3

The agreement between the numerical results and the analytical solution is now good. Comparison of the data for this case with the one computed on the five-point grid of Case 2 shows that grid refinement has reduced the F/D ratio from 5 to 1.25. The central differencing scheme seems to yield accurate results when the F/D ratio is low. The influence of the F/D ratio and the reasons for the appearance of 'wiggles' in central difference solutions when this ratio is high will be discussed below.

5.4 Properties of discretisation schemes

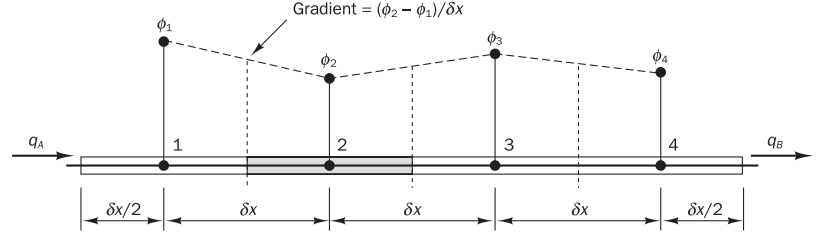
The failure of central differencing in certain cases involving combined convection and diffusion forces us to take a more in-depth look at the properties of discretisation schemes. In theory numerical results may be obtained that are indistinguishable from the 'exact' solution of the transport equation when the number of computational cells is infinitely large, irrespective of the differencing method used. However, in practical calculations we can only use a finite – sometimes quite small – number of cells, and our numerical results will only be physically realistic when the discretisation scheme has certain fundamental properties. The most important ones are:

- Conservativeness
- Boundedness
- Transportiveness

5.4.1 Conservativeness

Integration of the convection–diffusion equation over a finite number of control volumes yields a set of discretised conservation equations involving fluxes of the transported property ϕ through control volume faces. To ensure conservation of ϕ for the whole solution domain the flux of ϕ leaving a control volume across a certain face must be equal to the flux of ϕ entering the adjacent control volume through the same face. To achieve this the **flux** through a common face must be represented in a **consistent** manner – by one and the same expression – in adjacent control volumes.

For example, consider the one-dimensional steady state diffusion problem without source terms shown in Figure 5.7.

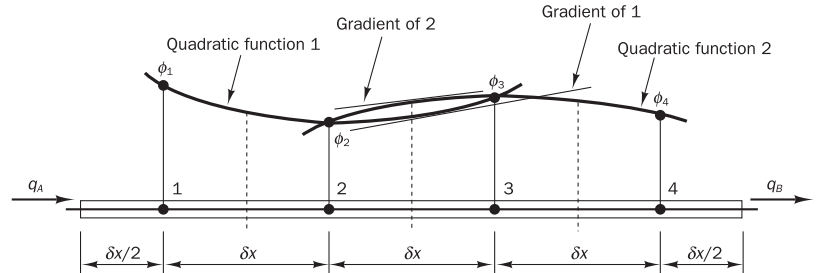
Figure 5.7 Example of consistent specification of diffusive fluxes

The fluxes across the domain boundaries are denoted by q_A and q_B. Let us consider four control volumes and apply central differencing to calculate the diffusive flux across the cell faces. The expression for the flux leaving the element around node 2 across its west face is $\Gamma_{w_2}(\phi_2 - \phi_1)/\delta x$ and the flux entering across its east face is $\Gamma_{e_2}(\phi_3 - \phi_2)/\delta x$. An overall flux balance may be obtained by summing the net flux through each control volume, taking into account the boundary fluxes for the control volumes around nodes 1 and 4:

$$\left[\Gamma_{e_1}\frac{(\phi_2 - \phi_1)}{\delta x} - q_A\right] + \left[\Gamma_{e_2}\frac{(\phi_3 - \phi_2)}{\delta x} - \Gamma_{w_2}\frac{(\phi_2 - \phi_1)}{\delta x}\right]$$
$$+ \left[\Gamma_{e_3}\frac{(\phi_4 - \phi_3)}{\delta x} - \Gamma_{w_3}\frac{(\phi_3 - \phi_2)}{\delta x}\right] + \left[q_B - \Gamma_{w_4}\frac{(\phi_4 - \phi_3)}{\delta x}\right]$$
$$= q_B - q_A \tag{5.21}$$

Since $\Gamma_{e_1} = \Gamma_{w_2}$, $\Gamma_{e_2} = \Gamma_{w_3}$ and $\Gamma_{e_3} = \Gamma_{w_4}$ the fluxes across control volume faces are expressed in a consistent manner and cancel out in pairs when summed over the entire domain. Only the two boundary fluxes q_A and q_B remain in the overall balance, so equation (5.21) expresses overall conservation of property ϕ. Flux consistency ensures conservation of ϕ over the entire domain for the central difference formulation of the diffusion flux.

Inconsistent flux interpolation formulae give rise to unsuitable schemes that do not satisfy overall conservation. For example, let us consider the situation where a quadratic interpolation formula, based on values at 1, 2 and 3, is used for control volume 2, and a quadratic profile, based on values at points 2, 3 and 4, is used for control volume 3.

As shown in Figure 5.8, the resulting quadratic profiles can be quite different.

Figure 5.8 Example of inconsistent specification of diffusive fluxes

Consequently, the flux values calculated at the east face of control volume 2 and the west face of control volume 3 may be unequal if the gradients of the

two curves are different at the cell face. If this is the case the two fluxes do not cancel out when summed and overall conservation is not satisfied. The example should not suggest to the reader that quadratic interpolation is entirely bad. Further on we will meet a quadratic discretisation practice – the so-called QUICK scheme – that *is* consistent.

5.4.2 Boundedness

The discretised equations at each nodal point represent a set of algebraic equations that needs to be solved. Normally iterative numerical techniques are used to solve large equation sets. These methods start the solution process from a guessed distribution of the variable ϕ and perform successive updates until a converged solution is obtained. Scarborough (1958) has shown that a **sufficient condition for a convergent iterative method** can be expressed in terms of the values of the coefficients of the discretised equations:

$$\boxed{\frac{\sum|a_{nb}|}{|a'_P|} \begin{cases} \leq 1 \text{ at all nodes} \\ < 1 \text{ at one node at least} \end{cases}} \tag{5.22}$$

Here a'_P is the net coefficient of the central node P (i.e. $a_P - S_P$), and the summation in the numerator is taken over all the neighbouring nodes (nb). If the differencing scheme produces coefficients that satisfy the above criterion the resulting matrix of coefficients is **diagonally dominant**. To achieve diagonal dominance we need large values of net coefficient ($a_P - S_P$) so the **linearisation** practice of **source terms** should ensure that $\boldsymbol{S_P}$ is always **negative**. If this is the case $-S_P$ is always positive and adds to a_P.

Diagonal dominance is a desirable feature for satisfying the 'boundedness' criterion. This states that in the **absence of sources** the internal nodal values of **property** ϕ should be **bounded by its boundary values**. Hence in a steady state conduction problem without sources and with boundary temperatures of 500°C and 200°C, all interior values of T should be less than 500°C and greater than 200°C. Another essential requirement for boundedness is that **all coefficients of the discretised equations should have the same sign** (usually all positive). Physically this implies that an increase in the variable ϕ at one node should result in an increase in ϕ at neighbouring nodes. If the discretisation scheme does not satisfy the boundedness requirements it is possible that the solution does not converge at all, or, if it does, that it contains 'wiggles'. This is powerfully illustrated by the results of Case 2 of Example 5.1. In all other worked examples we have developed discretised equations with positive coefficients a_P and a_{nb}, but in Case 2 most of the east coefficients were negative (see Table 5.3), and the solution contained large under- and overshoots!

5.4.3 Transportiveness

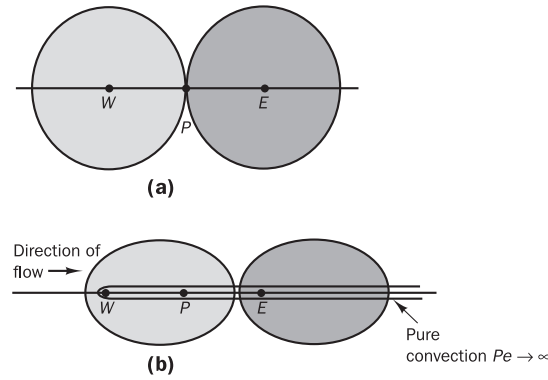
The transportiveness property of a fluid flow (Roache, 1976) can be illustrated by considering the effect at a point P due to two constant sources of ϕ at nearby points W and E on either side as shown in Figure 5.9. We define

the non-dimensional cell Peclet number as a measure of the relative strengths of convection and diffusion:

$$\boxed{Pe = \frac{F}{D} = \frac{\rho u}{\Gamma/\delta x}} \tag{5.23}$$

where δx = characteristic length (cell width)

The lines in Figure 5.9 indicate the general shape of contours of constant ϕ (say $\phi = 1$) due to both sources for different values of Pe. The value of ϕ at any point can be thought of as the sum of contributions due to the two sources.

Figure 5.9 Distribution of ϕ in the vicinity of two sources at different Peclet numbers: (a) pure convection, $Pe \to 0$; (b) diffusion and convection

Let us consider two extreme cases to identify the extent of the influence at node P due to the sources at W and E:

- no convection and pure diffusion ($Pe \to 0$)
- no diffusion and pure convection ($Pe \to \infty$)

In the case of pure diffusion the fluid is stagnant ($Pe \to 0$) and the contours of constant ϕ will be concentric circles centred around W and E since the diffusion process tends to spread ϕ equally in all directions. Figure 5.9a shows that both $\phi = 1$ contours pass through P, indicating that conditions at this point are influenced by both sources at W and E. As Pe increases the contours change shape from circular to elliptical and are shifted in the direction of the flow as shown in Figure 5.9b. Influencing becomes increasingly biased towards the upstream direction at large values of Pe, so, in the present case where the flow is in the positive x-direction, conditions at P will be mainly influenced by the upstream source at W. In the case of pure convection ($Pe \to \infty$) the elliptical contours are completely stretched out in the flow direction. All of property ϕ emanating from the sources at W and E is immediately transported downstream. Thus, conditions at P are now unaffected by the downstream source at E and completely dictated by the upstream source at W. Since there is no diffusion ϕ_P is equal to ϕ_W. If the flow is in the negative x-direction we would find that ϕ_P is equal to ϕ_E. It is very important that the relationship between the directionality of influencing and the flow direction and magnitude of the Peclet number, known as the **transportiveness**, is borne out in the discretisation scheme.

5.5 Assessment of the central differencing scheme for convection–diffusion problems

Conservativeness: The central differencing scheme uses consistent expressions to evaluate convective and diffusive fluxes at the control volume faces. The discussions in section 5.4.1 show that the scheme is conservative.

Boundedness:

(i) The internal coefficients of discretised scalar transport equation (5.14) are

a_W	a_E	a_P
$D_w + \frac{F_w}{2}$	$D_e - \frac{F_e}{2}$	$a_W + a_E + (F_e - F_w)$

A steady one-dimensional flow field is also governed by the discretised continuity equation (5.10). This equation states that $(F_e - F_w)$ is zero when the flow field satisfies continuity. Thus the expression for a_P in (5.14) becomes equal to $a_P = a_W + a_E$. The coefficients of the central differencing scheme satisfy the Scarborough criterion (5.22).

(ii) With $a_E = D_e - F_e/2$ the convective contribution to the east coefficient is negative; if the convection dominates it is possible for a_E to be negative. Given that $F_w > 0$ and $F_e > 0$ (i.e. the flow is unidirectional), for a_E to be positive D_e and F_e must satisfy the following condition:

$$F_e/D_e = Pe_e < 2 \tag{5.24}$$

If Pe_e is greater than 2 the east coefficient will be negative. This violates one of the requirements for boundedness and may lead to physically impossible solutions.

In the example of section 5.3 we took $Pe = 5$ in Case 2 so condition (5.24) is violated. The consequences were evident in the results, which showed large 'undershoots' and 'overshoots'. Taking Pe less than 2 in Cases 1 and 3 gave bounded answers close to the analytical solution.

Transportiveness: The central differencing scheme introduces influencing at node P from the directions of all its neighbours to calculate the convective and diffusive flux. Thus the scheme does not recognise the direction of the flow or the strength of convection relative to diffusion. It does not possess the transportiveness property at high Pe.

Accuracy: The Taylor series truncation error of the central differencing scheme is second-order (see Appendix A for further details). The requirement for positive coefficients in the central differencing scheme as given by formula (5.24) implies that the scheme will be stable and accurate only if $Pe = F/D < 2$. It is important to note that the cell Peclet number, as defined by (5.23), is a combination of fluid properties (ρ and Γ), a flow property (u) and a property of the computational grid (δx). So for given values of ρ and Γ it is only possible to satisfy condition (5.24) if the velocity is small, hence in diffusion-dominated low Reynolds number flows, or if the grid spacing is small. Owing to this limitation central differencing is not a suitable discretisation practice for general-purpose flow calculations. This creates the need for discretisation schemes which possess more favourable properties. Below we discuss the upwind, hybrid, power-law, QUICK and TVD schemes.

5.6 The upwind differencing scheme

One of the major inadequacies of the central differencing scheme is its inability to identify flow direction. The value of property ϕ at a west cell face is always influenced by both ϕ_P and ϕ_W in central differencing. In a strongly convective flow from west to east, the above treatment is unsuitable because the west cell face should receive much stronger influencing from node W than from node P. The upwind differencing or 'donor cell' differencing scheme takes into account the flow direction when determining the value at a cell face: the convected value of ϕ at a cell face is taken to be equal to the value at the upstream node. In Figure 5.10 we show the nodal values used to calculate cell face values when the flow is in the positive direction (west to east) and in Figure 5.11 those for the negative direction.

Figure 5.10

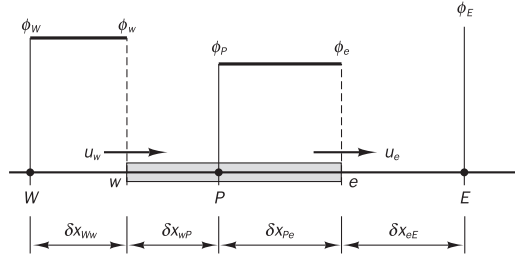
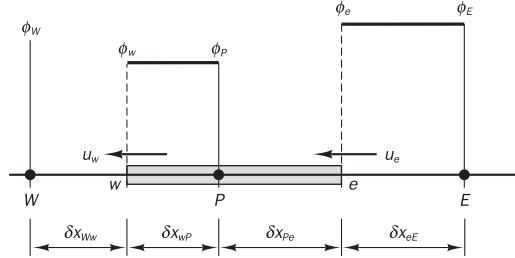



Figure 5.11

When the flow is in the positive direction, $u_w > 0$, $u_e > 0$ ($F_w > 0$, $F_e > 0$), the upwind scheme sets

$$\phi_w = \phi_W \quad \text{and} \quad \phi_e = \phi_P \tag{5.25}$$

and the discretised equation (5.9) becomes

$$F_e\phi_P - F_w\phi_W = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.26}$$

which can be rearranged as

$$(D_w + D_e + F_e)\phi_P = (D_w + F_w)\phi_W + D_e\phi_E$$

to give

$$[(D_w + F_w) + D_e + (F_e - F_w)]\phi_P = (D_w + F_w)\phi_W + D_e\phi_E \tag{5.27}$$

When the flow is in the negative direction, $u_w < 0$, $u_e < 0$ ($F_w < 0$, $F_e < 0$), the scheme takes

$$\phi_w = \phi_P \quad \text{and} \quad \phi_e = \phi_E \tag{5.28}$$

Now the discretised equation is

$$F_e\phi_E - F_w\phi_P = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.29}$$

or

$$[D_w + (D_e - F_e) + (F_e - F_w)]\phi_P = D_w\phi_W + (D_e - F_e)\phi_E \tag{5.30}$$

Identifying the coefficients of ϕ_W and ϕ_E as a_W and a_E, equations (5.27) and (5.30) can be written in the usual general form

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E} \tag{5.31}$$

with central coefficient

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

and neighbour coefficients

	a_W	a_E
$F_w > 0, F_e > 0$	$D_w + F_w$	D_e
$F_w < 0, F_e < 0$	D_w	$D_e - F_e$

A form of notation for the **neighbour coefficients of the upwind differencing method** that covers both flow directions is given below:

a_W	a_E
$D_w + \max(F_w, 0)$	$D_e + \max(0, -F_e)$

Example 5.2

Solve the problem considered in Example 5.1 using the upwind differencing scheme for (i) $u = 0.1$ m/s, (ii) $u = 2.5$ m/s with the coarse five-point grid.

Solution

The grid shown in Figure 5.3 is again used here for the discretisation. The discretisation equation at internal nodes 2, 3 and 4 and the relevant neighbour coefficients are given by (5.31) and its accompanying tables. Note that in this example $F = F_e = F_w = \rho u$ and $D = D_e = D_w = \Gamma/\delta x$ everywhere.

At the boundary node 1, the use of upwind differencing for the convective terms gives

$$F_e\phi_P - F_A\phi_A = D_e(\phi_E - \phi_P) - D_A(\phi_P - \phi_A) \tag{5.32}$$

And at node 5

$$F_B\phi_P - F_w\phi_W = D_B(\phi_B - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.33}$$

At the boundary nodes we have $D_A = D_B = 2\Gamma/\delta x = 2D$ and $F_A = F_B = F$, and as usual the boundary conditions enter the discretised equations as source contributions:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u} \tag{5.34}$$

with

$$\boxed{a_P = a_W + a_E + (F_e - F_w) - S_P}$$

and

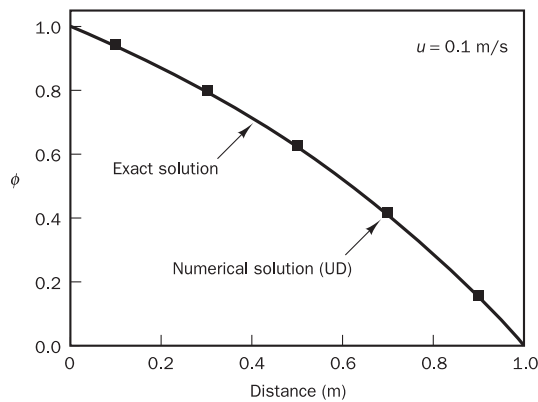
Node	a_W	a_E	S_P	S_u
1	0	D	$-(2D+F)$	$(2D+F)\phi_A$
2, 3, 4	$D+F$	D	0	0
5	$D+F$	0	$-2D$	$2D\phi_B$

The reader will by now be familiar with the process of calculating coefficients and constructing and solving the matrix equation. For the sake of brevity we leave this as an exercise and concentrate on the evaluation of the results. The analytical solution is again given by equation (5.15) and is compared with the numerical, upwind differencing, solution.

Case 1
$u = 0.1$ m/s: $F = \rho u = 0.1$, $D = \Gamma/\delta x = 0.1/0.2 = 0.5$ so $Pe = F/D = 0.2$. The results are summarised in Table 5.6, and Figure 5.12 shows that the upwind differencing (UD) scheme produces good results at this cell Peclet number.

Table 5.6

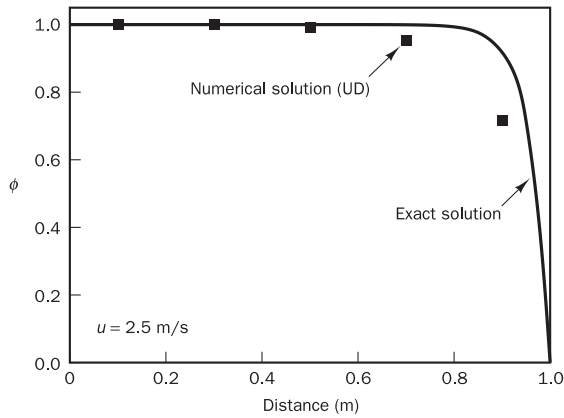
Node	Distance	Finite volume solution	Analytical solution	Difference	Percentage error
1	0.1	0.9337	0.9387	0.005	0.53
2	0.3	0.7879	0.7963	0.008	1.05
3	0.5	0.6130	0.6224	0.009	1.51
4	0.7	0.4031	0.4100	0.007	1.68
5	0.9	0.1512	0.1505	–0.001	–0.02

Figure 5.12 Comparison of the upwind difference numerical results and the analytical solution for Case 1

Case 2
$u = 2.5$ m/s: $F = \rho u = 2.5$, $D = \Gamma/\delta x = 0.1/0.2 = 0.5$ now $Pe = 5$. The numerical results are compared with the analytical solution in Table 5.7 and Figure 5.13.

The central differencing scheme failed to produce a reasonable result with the same grid resolution. The upwind scheme produces a much more realistic solution that is, however, not very close to the exact solution near boundary B.

Table 5.7

Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.1	0.9998	1.0000	0.0002	0.02
2	0.3	0.9987	0.9999	0.001	0.13
3	0.5	0.9921	0.9999	0.008	0.79
4	0.7	0.9524	0.9994	0.047	4.71
5	0.9	0.7143	0.9179	0.204	22.18

Figure 5.13 Comparison of the upwind difference numerical results and the analytical solution for Case 2

5.6.1 Assessment of the upwind differencing scheme

Conservativeness: The upwind differencing scheme utilises consistent expressions to calculate fluxes through cell faces: therefore it can be easily shown that the formulation is conservative.

Boundedness: The coefficients of the discretised equation are always positive and satisfy the requirements for boundedness. When the flow satisfies continuity the term $(F_e - F_w)$ in a_P (see (5.31)) is zero and gives $a_P = a_W + a_E$, which is desirable for stable iterative solutions. All the coefficients are positive and the coefficient matrix is diagonally dominant, hence no 'wiggles' occur in the solution.

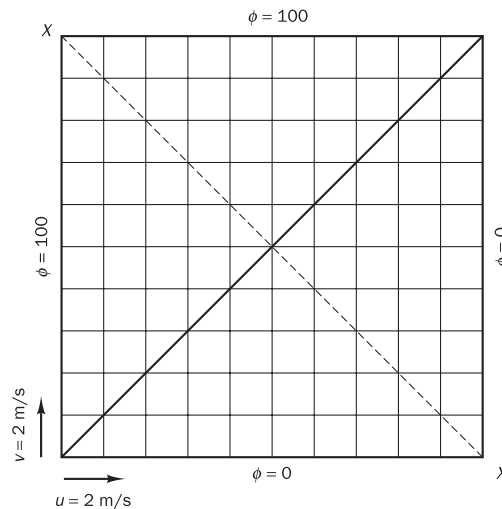
Transportiveness: The scheme accounts for the direction of the flow so transportiveness is built into the formulation.

Accuracy: The scheme is based on the backward differencing formula so the accuracy is only first-order on the basis of the Taylor series truncation error (see Appendix A).

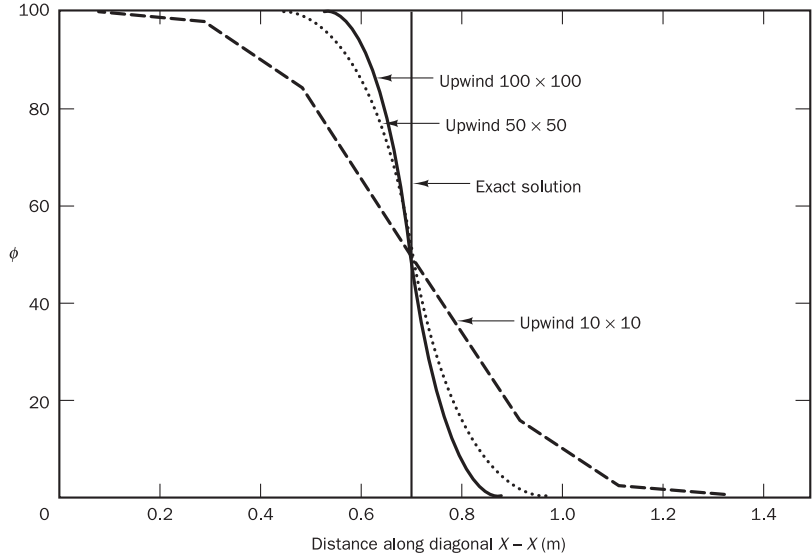
Because of its simplicity the upwind differencing scheme has been widely applied in early CFD calculations. It can be easily extended to

multi-dimensional problems by repeated application of the upwind strategy embodied in the coefficients of (5.31) in each co-ordinate direction. A major drawback of the scheme is that it produces erroneous results when the flow is not aligned with the grid lines. The upwind differencing scheme causes the distributions of the transported properties to become smeared in such problems. The resulting error has a diffusion-like appearance and is referred to as **false diffusion**. The effect can be illustrated by calculating the transport of scalar property ϕ using upwind differencing in a domain where the flow is at an angle to a Cartesian grid.

In Figure 5.14 we have a domain where $u = v = 2$ m/s everywhere so the velocity field is uniform and parallel to the diagonal (solid line) across the grid. The boundary conditions for the scalar are $\phi = 0$ along the south and east boundaries, and $\phi = 100$ on the west and north boundaries. At the first and the last nodes where the diagonal intersects the boundary a value of 50 is assigned to property ϕ.

Figure 5.14 Flow domain for the illustration of false diffusion

To identify the false diffusion due to the upwind scheme, a pure convection process is considered without physical diffusion. There are no source terms for ϕ and a steady state solution is sought. The correct solution is known in this case. As the flow is parallel to the solid diagonal the value of ϕ at all nodes above the diagonal should be 100 and below the diagonal it should be zero. The degree of false diffusion can be illustrated by calculating the distribution of ϕ and plotting the results along the diagonal (*X*–*X*). Since there is no physical diffusion the exact solution exhibits a step change of ϕ from 100 to zero when the diagonal *X*–*X* crosses the solid diagonal. The calculated results for different grids are shown in Figure 5.15 together with the exact solution. The numerical results show badly smeared profiles.

The error is largest for the coarsest grid, and the figure shows that refinement of the grid can, in principle, overcome the problem of false diffusion. The results for 50 × 50 and 100 × 100 grids show profiles that are closer to the exact solution. In practical flow calculations, however, the degree of grid refinement required to eliminate false diffusion can be prohibitively expensive. Trials have shown that, in high Reynolds number

Figure 5.15

flows, false diffusion can be large enough to give physically incorrect results (Leschziner, 1980; Huang *et al.*, 1985). Therefore, the upwind differencing scheme is not entirely suitable for accurate flow calculations and considerable research has been directed towards finding improved discretisation schemes.

5.7 The hybrid differencing scheme

The hybrid differencing scheme of Spalding (1972) is based on a combination of central and upwind differencing schemes. The central differencing scheme, which is second-order accurate, is employed for small Peclet numbers ($Pe < 2$) and the upwind scheme, which is first-order accurate but accounts for transportiveness, is employed for large Peclet numbers ($Pe \geq 2$). As before, we develop the discretisation of the one-dimensional convection–diffusion equation without source terms. This equation can be interpreted as a flux balance equation. The hybrid differencing scheme uses piecewise formulae based on the local Peclet number to evaluate the net flux through each control volume face. The Peclet number is evaluated at the face of the control volume. For example, for a west face,

$$Pe_w = \frac{F_w}{D_w} = \frac{(\rho u)_w}{\Gamma_w / \delta x_{WP}} \tag{5.35}$$

The hybrid differencing formula for the net flux per unit area through the west face is as follows:

$$\begin{aligned} q_w &= F_w \left[\frac{1}{2}\left(1 + \frac{2}{Pe_w}\right)\phi_W + \frac{1}{2}\left(1 - \frac{2}{Pe_w}\right)\phi_P \right] && \text{for } -2 < Pe_w < 2 \\ q_w &= F_w \phi_W && \text{for } Pe_w \geq 2 \\ q_w &= F_w \phi_P && \text{for } Pe_w \leq -2 \end{aligned} \tag{5.36}$$

It can be easily seen that for low Peclet numbers this is equivalent to using central differencing for the convection and diffusion terms, but when $|Pe| > 2$ it is equivalent to upwinding for convection and setting the diffusion to zero. The general form of the discretised equation is

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E} \tag{5.37}$$

The central coefficient is given by

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

After some rearrangement it is easy to verify that the neighbour coefficients for the **hybrid differencing** scheme for steady **one-dimensional convection–diffusion** can be written as follows:

a_W	a_E
$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$

Example 5.3

Solve the problem considered in Case 2 of Example 5.1 using the hybrid scheme for u = 2.5 m/s. Compare a 5-node solution with a 25-node solution.

Solution

If we use the 5-node grid and the data of Case 2 of Example 5.1 and u = 2.5 m/s we have: $F = F_e = F_w = \rho u = 2.5$ and $D = D_e = D_w = \Gamma/\delta x = 0.5$ and hence a Peclet number $Pe_w = Pe_e = \rho u \delta x/\Gamma = 5$. Since the cell Peclet number Pe is greater than 2 the hybrid scheme uses the upwind expression for the convective terms and sets the diffusion to zero.

The discretised equation at internal nodes 2, 3 and 4 is defined by (5.37) and its coefficients. We also need to introduce boundary conditions at nodes 1 and 5, which need special treatment. At the boundary node 1 we write

$$F_e\phi_P - F_A\phi_A = 0 - D_A(\phi_P - \phi_A) \tag{5.38}$$

and at node 5

$$F_B\phi_P - F_w\phi_W = D_B(\phi_B - \phi_P) - 0 \tag{5.39}$$

It can be seen that the diffusive flux at the boundary is entered on the right hand side and the convective fluxes are given by means of the upwind method. We note that $F_A = F_B = F$ and $D_B = 2\Gamma/\delta x = 2D$ so the discretised equation can be written as

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u} \tag{5.40}$$

with

$$\boxed{a_P = a_W + a_E + (F_e - F_w) - S_P}$$

and

Node	a_W	a_E	S_P	S_u
1	0	0	$-(2D + F)$	$(2D + F)\phi_A$
2,3,4	F	0	0	0
5	F	0	$-2D$	$2D\phi_B$

Substitution of numerical values gives the coefficients summarised in Table 5.8.

Table 5.8

Node	a_W	a_E	S_u	S_P	$a_P = a_W + a_E - S_P$
1	0	0	$3.5\phi_A$	−3.5	3.5
2	2.5	0	0	0	2.5
3	2.5	0	0	0	2.5
4	2.5	0	0	0	2.5
5	2.5	0	$1.0\phi_B$	−1.0	3.5

The matrix form of the equation set is

$$\begin{bmatrix} 3.5 & 0 & 0 & 0 & 0 \\ -2.5 & 2.5 & 0 & 0 & 0 \\ 0 & -2.5 & 2.5 & 0 & 0 \\ 0 & 0 & -2.5 & 2.5 & 0 \\ 0 & 0 & 0 & -2.5 & 3.5 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 3.5 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.41}$$

The solution to the above system is

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 0.7143 \end{bmatrix} \tag{5.42}$$

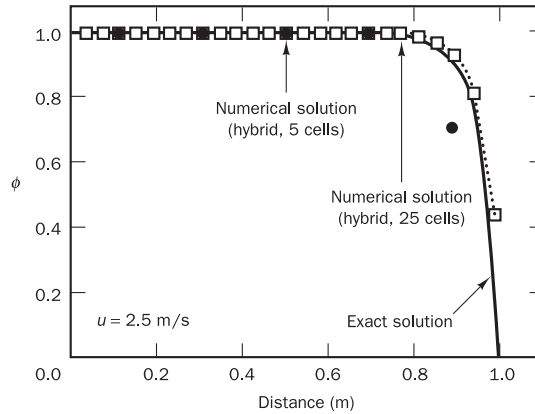
Comparison with the analytical solution

The numerical results are compared with the analytical solution in Table 5.9 and, since the cell Peclet number is high, they are the same as those for pure

Table 5.9

Node	*Distance*	*Finite volume solution*	*Analytical solution*	*Difference*	*Percentage error*
1	0.1	1.0	1.0000	0.0	0.0
2	0.3	1.0	0.9999	−0.0001	−0.01
3	0.5	1.0	0.9999	−0.0001	−0.01
4	0.7	1.0	0.9994	−0.0006	−0.06
5	0.9	0.7143	0.9179	0.204	22.18

upwind differencing. When the grid is refined to an extent that the cell $Pe < 2$, the scheme reverts to central differencing and produces an accurate solution. This illustrated by using a 25-node grid with $\delta x = 0.04$ m so $F = D = 2.5$. The results computed on both the coarse and the fine grids are shown in Figure 5.16 together with the analytical solution. Now $Pe = 1$, the hybrid scheme reverts to central differencing, and it can be seen that the solution obtained with the fine grid is remarkably good.

Figure 5.16

5.7.1 Assessment of the hybrid differencing scheme

The hybrid difference scheme exploits the favourable properties of the upwind and central differencing schemes. It switches to upwind differencing when central differencing produces inaccurate results at high Pe numbers. The scheme is fully conservative and since the coefficients are always positive it is unconditionally bounded. It satisfies the transportiveness requirement by using an upwind formulation for large values of Peclet number. The scheme produces physically realistic solutions and is highly stable when compared with higher-order schemes such as QUICK to be discussed later in the chapter. Hybrid differencing has been widely used in various CFD procedures and has proved to be very useful for predicting practical flows. The disadvantage is that the accuracy in terms of Taylor series truncation error is only first-order.

5.7.2 Hybrid differencing scheme for multi-dimensional convection–diffusion

The hybrid differencing scheme can easily be extended to two- and three-dimensional problems by repeated application of the derivation in each new co-ordinate direction. The discretised equation that covers all cases is given by

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + a_B\phi_B + a_T\phi_T} \tag{5.43}$$

with central coefficient

$$\boxed{a_P = a_W + a_E + a_S + a_N + a_B + a_T + \Delta F}$$

and the coefficients of this equation for the **hybrid differencing scheme** are as follows:

	One-dimensional flow	*Two-dimensional flow*	*Three-dimensional flow*
a_W	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$
a_E	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$
a_S	–	$\max\left[F_s, \left(D_s + \frac{F_s}{2}\right), 0\right]$	$\max\left[F_s, \left(D_s + \frac{F_s}{2}\right), 0\right]$
a_N	–	$\max\left[-F_n, \left(D_n - \frac{F_n}{2}\right), 0\right]$	$\max\left[-F_n, \left(D_n - \frac{F_n}{2}\right), 0\right]$
a_B	–	–	$\max\left[F_b, \left(D_b + \frac{F_b}{2}\right), 0\right]$
a_T	–	–	$\max\left[-F_t, \left(D_t - \frac{F_t}{2}\right), 0\right]$
ΔF	$F_e - F_w$	$F_e - F_w + F_n - F_s$	$F_e - F_w + F_n - F_s + F_t - F_b$

In the above expressions the values of F and D are calculated with the following formulae:

Face	*w*	*e*	*s*	*n*	*b*	*t*
F	$(\rho u)_w A_w$	$(\rho u)_e A_e$	$(\rho v)_s A_s$	$(\rho v)_n A_n$	$(\rho w)_b A_b$	$(\rho w)_t A_t$
D	$\frac{\Gamma_w}{\delta x_{WP}} A_w$	$\frac{\Gamma_e}{\delta x_{PE}} A_e$	$\frac{\Gamma_s}{\delta y_{SP}} A_s$	$\frac{\Gamma_n}{\delta y_{PN}} A_n$	$\frac{\Gamma_b}{\delta x_{BP}} A_b$	$\frac{\Gamma_t}{\delta x_{PT}} A_t$

Modifications to these coefficients to cater for boundary conditions in two and three dimensions are available in the form of expressions such as (5.40).

5.8 The power-law scheme

The power-law differencing scheme of Patankar (1980) is a more accurate approximation to the one-dimensional exact solution and produces better results than the hybrid scheme. In this scheme diffusion is set to zero when

cell Pe exceeds 10. If $0 < Pe < 10$ the flux is evaluated by using a polynomial expression. For example, the net flux per unit area at the west control volume face is evaluated using

$$q_w = F_w[\phi_W - \beta_w(\phi_P - \phi_W)] \text{ for } 0 < Pe < 10 \quad (5.44a)$$

$$\text{where } \beta_w = (1 - 0.1Pe_w)^5/Pe_w$$

and

$$q_w = F_w\phi_W \text{ for } Pe > 10 \quad (5.44b)$$

The coefficients of the one-dimensional discretised equation utilising the **power-law** scheme for steady **one-dimensional convection–diffusion** are given by

$$\text{central coefficient: } a_P = a_W + a_E + (F_e - F_w)$$

and

a_W	a_E
$D_w \max[0, (1 - 0.1\lvert Pe_w\rvert)^5] + \max[F_w, 0]$	$D_e \max[0, (1 - 0.1\lvert Pe_e\rvert)^5] + \max[-F_e, 0]$

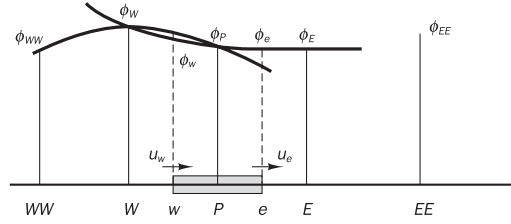
Properties of the power-law differencing scheme are similar to those of the hybrid scheme. The power-law differencing scheme is more accurate for one-dimensional problems since it attempts to represent the exact solution more closely. The scheme has proved to be useful in practical flow calculations and can be used as an alternative to the hybrid scheme. In some commercial computer codes, e.g. FLUENT version 6.2, this scheme is available as a discretisation option for the user to choose (FLUENT documentation, 2006).

5.9 Higher-order differencing schemes for convection–diffusion problems

The accuracy of hybrid and upwind schemes is only first-order in terms of Taylor series truncation error (TSTE). The use of upwind quantities ensures that the schemes are very stable and obey the transportiveness requirement, but the first-order accuracy makes them prone to numerical diffusion errors. Such errors can be minimised by employing higher-order discretisation. Higher-order schemes involve more neighbour points and reduce the discretisation errors by bringing in a wider influence. The central differencing scheme, which has second-order accuracy, proved to be unstable and does not possess the transportiveness property. Formulations that do not take into account the flow direction are unstable and, therefore, more accurate higher-order schemes, which preserve upwinding for stability and sensitivity to the flow direction, are needed. Below we discuss in some detail Leonard's QUICK scheme, which is the oldest of these higher-order schemes.

5.9.1 Quadratic upwind differencing scheme: the QUICK scheme

The quadratic upstream interpolation for convective kinetics (QUICK) scheme of Leonard (1979) uses a three-point upstream-weighted quadratic interpolation for cell face values. The face value of ϕ is obtained from a quadratic function passing through two bracketing nodes (on each side of the face) and a node on the upstream side (Figure 5.17).

Figure 5.17 Quadratic profiles used in the QUICK scheme

For example, when $u_w > 0$ and $u_e > 0$ a quadratic fit through WW, W and P is used to evaluate ϕ_w, and a further quadratic fit through W, P and E to calculate ϕ_e. For $u_w < 0$ and $u_e < 0$ values of ϕ at W, P and E are used for ϕ_w, and values at P, E and EE for ϕ_e. It can be shown that for a uniform grid the value of ϕ at the cell face between two bracketing nodes i and $i - 1$ and upstream node $i - 2$ is given by the following formula:

$$\phi_{face} = \frac{6}{8}\phi_{i-1} + \frac{3}{8}\phi_i - \frac{1}{8}\phi_{i-2} \tag{5.45}$$

When $u_w > 0$, the bracketing nodes for the west face w are W and P, the upstream node is WW (Figure 5.17) and

$$\phi_w = \frac{6}{8}\phi_W + \frac{3}{8}\phi_P - \frac{1}{8}\phi_{WW} \tag{5.46}$$

When $u_e > 0$, the bracketing nodes for the east face e are P and E, the upstream node is W, so

$$\phi_e = \frac{6}{8}\phi_P + \frac{3}{8}\phi_E - \frac{1}{8}\phi_W \tag{5.47}$$

The diffusion terms may be evaluated using the gradient of the approximating parabola. It is interesting to note that on a uniform grid this practice gives the same expressions as central differencing for diffusion, since the slope of the chord between two points on a parabola is equal to the slope of the tangent to the parabola at its midpoint. If $F_w > 0$ and $F_e > 0$, and if we use equations (5.46)–(5.47) for the convective terms and central differencing for the diffusion terms, the discretised form of the one-dimensional convection–diffusion transport equation (5.9) may be written as

$$\left[F_e\left(\frac{6}{8}\phi_P + \frac{3}{8}\phi_E - \frac{1}{8}\phi_W\right) - F_w\left(\frac{6}{8}\phi_W + \frac{3}{8}\phi_P - \frac{1}{8}\phi_{WW}\right)\right]$$
$$= D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)$$

which can be rearranged to give

$$\left[D_w - \frac{3}{8}F_w + D_e + \frac{6}{8}F_e\right]\phi_P = \left[D_w + \frac{6}{8}F_w + \frac{1}{8}F_e\right]\phi_W$$
$$+ \left[D_e - \frac{3}{8}F_e\right]\phi_E - \frac{1}{8}F_w\phi_{WW} \tag{5.48}$$

This is now written in the standard form for discretised equations:

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_{WW}\phi_{WW} \tag{5.49}$$

where

a_W	a_E	a_{WW}	a_P
$D_w + \frac{6}{8}F_w + \frac{1}{8}F_e$	$D_e - \frac{3}{8}F_e$	$-\frac{1}{8}F_w$	$a_W + a_E + a_{WW} + (F_e - F_w)$

For $F_w < 0$ and $F_e < 0$ the flux across the west and east boundaries is given by the expressions

$$\phi_w = \frac{6}{8}\phi_P + \frac{3}{8}\phi_W - \frac{1}{8}\phi_E \tag{5.50}$$

$$\phi_e = \frac{6}{8}\phi_E + \frac{3}{8}\phi_P - \frac{1}{8}\phi_{EE}$$

Substitution of these two formulae for the convective terms in the discretised convection–diffusion equation (5.9) together with central differencing for the diffusion terms leads, after rearrangement as above, to the following coefficients:

a_W	a_E	a_{EE}	a_P
$D_w + \frac{3}{8}F_w$	$D_e - \frac{6}{8}F_e - \frac{1}{8}F_w$	$\frac{1}{8}F_e$	$a_W + a_E + a_{EE} + (F_e - F_w)$

General expressions, valid for positive and negative flow directions, can be obtained by combining the two sets of coefficients above.

The **QUICK scheme for one-dimensional convection–diffusion problems** can be summarised as follows:

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_{WW}\phi_{WW} + a_{EE}\phi_{EE} \tag{5.51}$$

with central coefficient

$$a_P = a_W + a_E + a_{WW} + a_{EE} + (F_e - F_w)$$

and neighbour coefficients

a_W	a_{WW}	a_E	a_{EE}
$D_w + \frac{6}{8}\alpha_w F_w + \frac{1}{8}\alpha_e F_e + \frac{3}{8}(1 - \alpha_w)F_w$	$-\frac{1}{8}\alpha_w F_w$	$D_e - \frac{3}{8}\alpha_e F_e - \frac{6}{8}(1 - \alpha_e)F_e - \frac{1}{8}(1 - \alpha_w)F_w$	$\frac{1}{8}(1 - \alpha_e)F_e$

where

$$\alpha_w = 1 \text{ for } F_w > 0 \text{ and } \alpha_e = 1 \text{ for } F_e > 0$$
$$\alpha_w = 0 \text{ for } F_w < 0 \text{ and } \alpha_e = 0 \text{ for } F_e < 0$$

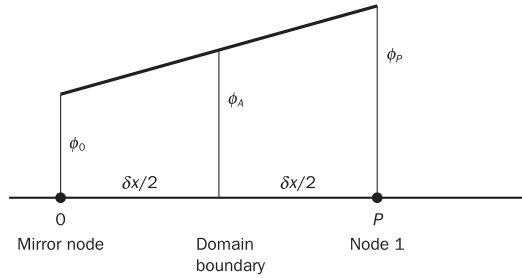
Example 5.4

Using the QUICK scheme solve the problem considered in Example 5.1 for u = 0.2 m/s on a five-point grid. Compare the QUICK solution with the exact and the central differencing solution.

Solution

As before, the five-node grid introduced in Example 5.1 is used for the discretisation. With the data of this example and $u = 0.2$ m/s we have $F = F_e = F_w = 0.2$ and $D = D_e = D_w = 0.5$ everywhere so that the cell Peclet number becomes $Pe_w = Pe_e = \rho u \delta x/\Gamma = 0.4$. The discretisation equation with the QUICK scheme at internal nodes 3 and 4 is given by (5.51) together with its coefficients.

In the QUICK scheme the ϕ-value at cell boundaries is calculated with formulae (5.46)–(5.47) that use three nodal values. Nodes 1, 2 and 5 are all affected by the proximity of domain boundaries and need to be treated separately. At the boundary node 1, ϕ is given at the west (w) face ($\phi_w = \phi_A$), but there is no west (W) node to evaluate ϕ_e at the east face by (5.47). To overcome this problem Leonard (1979) suggested a linear extrapolation to create a 'mirror' node at a distance $\delta x/2$ to the west of the physical boundary. This is illustrated in Figure 5.18.

Figure 5.18 Mirror node treatment at the boundary

It can be easily shown that the linearly extrapolated value at the mirror node is given by

$$\phi_0 = 2\phi_A - \phi_P \tag{5.52}$$

The extrapolation to the 'mirror' node has given us the required W node for the formula (5.47) that calculates ϕ_e at the east face of control volume 1:

$$\begin{aligned}\phi_e &= \frac{6}{8}\phi_P + \frac{3}{8}\phi_E - \frac{1}{8}(2\phi_A - \phi_P) \\ &= \frac{7}{8}\phi_P + \frac{3}{8}\phi_E - \frac{2}{8}\phi_A \end{aligned} \tag{5.53}$$

At the boundary nodes the gradients must be evaluated using an expression consistent with formula (5.53). It can be shown that the diffusive flux through the west boundary is given by

$$\Gamma \frac{\partial \phi}{\partial x}\bigg|_A = \frac{D_A^*}{3}(9\phi_P - 8\phi_A - \phi_E) \tag{5.54}$$

where $D_A^* = \dfrac{\Gamma}{\delta x}$

The superscript * is used to indicate that, in the QUICK scheme, the diffusion conductances at boundary nodes and interior nodes have the same value, i.e. $D_A^* = D = \Gamma/\delta x$. This aspect is different from the discretisation schemes we have discussed thus far. These used the half-cell approximation, so the diffusive conductance at the boundary cell was always $D_A = 2D = 2\Gamma/\delta x$.

The discretised equation at node 1 is

$$F_e\left[\frac{7}{8}\phi_P + \frac{3}{8}\phi_E - \frac{2}{8}\phi_A\right] - F_A\phi_A = D_e(\phi_E - \phi_P) - \frac{D_A^*}{3}(9\phi_P - 8\phi_A - \phi_E) \tag{5.55}$$

At control volume 5, the ϕ-value at the east face is known ($\phi_e = \phi_B$) and the diffusive flux of ϕ through the east boundary is given by

$$\Gamma \frac{\partial \phi}{\partial x}\bigg|_B = \frac{D_B^*}{3}(8\phi_B - 9\phi_P + \phi_W) \tag{5.56}$$

where $D_B^* = \dfrac{\Gamma}{\delta x}$

At node 5 the discretised equation becomes

$$F_B\phi_B - F_w\left[\frac{6}{8}\phi_W + \frac{3}{8}\phi_P - \frac{1}{8}\phi_{WW}\right] = \frac{D_B^*}{3}(8\phi_B - 9\phi_P + \phi_W) - D_w(\phi_P - \phi_W) \tag{5.57}$$

Since a special expression is used to evaluate ϕ at the east face of control volume 1 we must use the same expression for ϕ to calculate the convective flux through the west face of control volume 2 to ensure flux consistency. So at node 2 we have

$$F_e\left[\frac{6}{8}\phi_P + \frac{3}{8}\phi_E - \frac{1}{8}\phi_W\right] - F_w\left[\frac{7}{8}\phi_W + \frac{3}{8}\phi_P - \frac{2}{8}\phi_A\right] = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.58}$$

The discretised equations for nodes 1, 2 and 5 are now written to fit into the standard form to give

$$a_P\phi_P = a_{WW}\phi_{WW} + a_W\phi_W + a_E\phi_E + S_u \tag{5.59}$$

with

$$a_P = a_{WW} + a_W + a_E + (F_e - F_w) - S_P$$

and

Node	a_{WW}	a_W	a_E	S_P	S_u
1	0	0	$D_e + \frac{1}{3}D_A^* - \frac{3}{8}F_e$	$-\left(\frac{8}{3}D_A^* + \frac{2}{8}F_e + F_A\right)$	$\left(\frac{8}{3}D_A^* + \frac{2}{8}F_e + F_A\right)\phi_A$
2	0	$D_w + \frac{7}{8}F_w + \frac{1}{8}F_e$	$D_e - \frac{3}{8}F_e$	$\frac{1}{4}F_w$	$-\frac{1}{4}F_w\phi_A$
5	$-\frac{1}{8}F_w$	$D_w + \frac{1}{3}D_B^* + \frac{6}{8}F_w$	0	$-\left(\frac{8}{3}D_B^* - F_B\right)$	$\left(\frac{8}{3}D_B^* - F_B\right)\phi_B$

Substitution of numerical values gives the coefficients summarised in Table 5.10.

Table 5.10

Node	a_W	a_E	a_{WW}	S_u	S_P	a_P
1	0	0.592	0	$1.583\phi_A$	−1.583	2.175
2	0.7	0.425	0	$-0.05\phi_A$	0.05	1.075
3	0.675	0.425	−0.025	0	0	1.075
4	0.675	0.425	−0.025	0	0	1.075
5	0.817	0	−0.025	$1.133\phi_B$	−1.133	1.925

The matrix form of the equation set is

$$\begin{bmatrix} 2.175 & -0.592 & 0 & 0 & 0 \\ -0.7 & 1.075 & -0.425 & 0 & 0 \\ 0.025 & -0.675 & 1.075 & -0.425 & 0 \\ 0 & 0.025 & -0.675 & 1.075 & -0.425 \\ 0 & 0 & 0.025 & -0.817 & 1.925 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 1.583 \\ -0.05 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (5.60)$$

The solution to the above system is

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 0.9648 \\ 0.8707 \\ 0.7309 \\ 0.5226 \\ 0.2123 \end{bmatrix} \quad (5.61)$$

Comparison with the analytical solution

Figure 5.19 shows that the QUICK solution is almost indistinguishable from the exact solution. Table 5.11 confirms that the errors are very small even with this coarse mesh. Following the steps outlined in Example 5.1 the central differencing solution is computed with the data given above. The sum of absolute errors in Table 5.11 indicates that the QUICK scheme gives a more accurate solution than the central differencing scheme.

Figure 5.19 Comparison of QUICK solution with the analytical solution

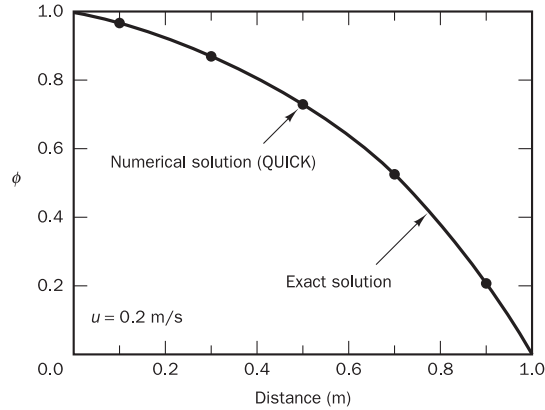



Table 5.11

Node	*Distance*	*Analytical solution*	*QUICK solution*	*Difference*	*CD solution*	*Difference*
1	0.1	0.9653	0.9648	0.0005	0.9696	0.0043
2	0.3	0.8713	0.8707	0.0006	0.8786	0.0073
3	0.5	0.7310	0.7309	0.0001	0.7421	0.0111
4	0.7	0.5218	0.5226	–0.0008	0.5374	0.0156
5	0.9	0.2096	0.2123	–0.0027	0.2303	0.0207
∑ Absolute error				0.0047		0.059

5.9.2 Assessment of the QUICK scheme

The scheme uses consistent quadratic profiles – the cell face values of fluxes are always calculated by quadratic interpolation between two bracketing nodes and an upstream node – and is therefore conservative. Since the scheme is based on a quadratic function its accuracy in terms of Taylor series truncation error is third-order on a uniform mesh. The transportiveness property is built into the scheme as the quadratic function is based on two upstream and one downstream nodal values. If the flow field satisfies continuity the coefficient a_P equals the sum of all neighbour coefficients, which is desirable for boundedness.

On the downside, the main coefficients (E and W) are not guaranteed to be positive and the coefficients a_{WW} and a_{EE} are negative. For example, if $u_w > 0$ and $u_e > 0$ the east coefficient becomes negative at relatively modest cell Peclet numbers ($Pe_e = F_e/D_e > 8/3$). This gives rise to stability problems and unbounded solutions under certain flow conditions. Similarly the west coefficient can become negative when the flow is in the negative direction. The QUICK scheme is therefore conditionally stable.

Another notable feature is the fact that the discretised equations involve not only immediate-neighbour nodes but also nodes further away. Tri-diagonal matrix solution methods (see Chapter 7) are not directly applicable.

5.9.3 Stability problems of the QUICK scheme and remedies

Since the QUICK scheme in the form presented above can be unstable due to the appearance of negative main coefficients, it has been reformulated in different ways that alleviate stability problems. These formulations all involve placing troublesome negative coefficients in the source term so as to retain positive main coefficients. The contributing part is appropriately weighted to give better stability and positive coefficients as far as possible. Some of the better known practical approaches are described in Han *et al.* (1981), Pollard and Siu (1982) and Hayase *et al.* (1992). The last authors generalised the approach for rearranging QUICK schemes and derived a stable and fast converging variant.

The **Hayase *et al.* (1992) QUICK scheme** can be summarised as follows:

$$\phi_w = \phi_W + \frac{1}{8}[3\phi_P - 2\phi_W - \phi_{WW}] \qquad \text{for } F_w > 0$$

$$\phi_e = \phi_P + \frac{1}{8}[3\phi_E - 2\phi_P - \phi_W] \qquad \text{for } F_e > 0 \tag{5.62}$$

$$\phi_w = \phi_P + \frac{1}{8}[3\phi_W - 2\phi_P - \phi_E] \qquad \text{for } F_w < 0$$

$$\phi_e = \phi_E + \frac{1}{8}[3\phi_P - 2\phi_E - \phi_{EE}] \qquad \text{for } F_e < 0$$

The discretisation equation takes the form

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + \bar{S}} \tag{5.63}$$

The central coefficient is

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

and

a_W	a_E	$\bar{S}$
$D_w + \alpha_w F_w$	$D_e - (1 - \alpha_e)F_e$	$\frac{1}{8}(3\phi_P - 2\phi_W - \phi_{WW})\alpha_w F_w$ $+ \frac{1}{8}(\phi_W + 2\phi_P - 3\phi_E)\alpha_e F_e$ $+ \frac{1}{8}(3\phi_W - 2\phi_P - \phi_E)(1 - \alpha_w)F_w$ $+ \frac{1}{8}(2\phi_E + \phi_{EE} - 3\phi_P)(1 - \alpha_e)F_e$

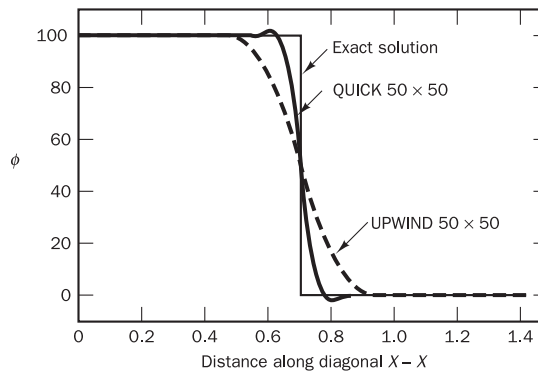
where

$\alpha_w = 1$ for $F_w > 0$ and $\alpha_e = 1$ for $F_e > 0$
$\alpha_w = 0$ for $F_w < 0$ and $\alpha_e = 0$ for $F_e < 0$

The advantage of this approach is that the main coefficients are positive and satisfy the requirements for conservativeness, boundedness and transportiveness. The allocation to the source term of the part of the discretisation that contains negative coefficients is called **deferred correction** and relies on the scheme being applied as part of an iterative loop structure. At the nth iteration the source term is evaluated using values known at the end of the previous $(n - 1)$th iteration, i.e. 'correction' of the main coefficients is 'deferred' by one iteration. After a sufficiently large number of iterations the correction 'catches up' with the rest of the solution, so all variations of QUICK, including the one developed by Hayase *et al.*, will give the same converged solution.

5.9.4 General comments on the QUICK differencing scheme

The QUICK differencing scheme has greater formal accuracy than the central differencing or hybrid schemes, and it retains the upwind-weighted characteristics. The resultant false diffusion is small, and solutions achieved with coarse grids are often considerably more accurate than those of the upwind or hybrid schemes. Figure 5.20 shows a comparison between upwind and QUICK for the two-dimensional test case considered in section 5.6.1. It can be seen that the QUICK scheme matches the exact solution much more accurately than the upwind scheme on a 50 × 50 grid.

Figure 5.20 Comparison of QUICK and upwind solutions for the 2D test case considered in section 5.6.1

The QUICK scheme can, however, give (minor) undershoots and overshoots, as is evident in Figure 5.20. In complex flow calculations, the use of QUICK can lead to subtle problems caused by such unbounded results: for example, they could give rise to negative turbulence kinetic energy (k) in k–ε model (see Chapter 3) computations. The possibility of undershoots and overshoots needs to be considered when interpreting solutions.

5.10 TVD schemes

Schemes of third-order and above have been developed for the discretisation of convective terms with varying degrees of success. Implementation of boundary conditions can be problematic with such higher-order schemes. The fact that the QUICK scheme and other higher-order schemes can give undershoots and overshoots has led to the development of second-order

schemes that avoid these problems. The class of TVD (total variation diminishing) schemes has been specially formulated to achieve oscillation-free solutions and has proved to be useful in CFD calculations. TVD is a property used in the discretisation of equations governing time-dependent gas dynamics problems. More recently, schemes with this property have also become popular in general-purpose CFD solvers. Fundamentals of the development of TVD methodology involves a fair amount of mathematical background. However, the ideas behind TVD schemes can be easily illustrated in the context of the discretisation practices presented in the previous sections by considering the basic properties of standard schemes and their deficiencies.

As discussed earlier, the basic upwind differencing scheme is the most stable and unconditionally bounded scheme, but it introduces a high level of false diffusion due to its low order of accuracy (first-order). Higher-order schemes such as central differencing and QUICK can give spurious oscillations or 'wiggles' when the Peclet number is high. When such higher-order schemes are used to solve for turbulent quantities, e.g. turbulence energy and rate of dissipation, wiggles can give physically unrealistic negative values and instability. TVD schemes are designed to address this undesirable oscillatory behaviour of higher-order schemes. In TVD schemes the tendency towards oscillation is counteracted by adding an artificial diffusion fragment or by adding a weighting towards upstream contribution. In the literature early schemes based on these ideas were called flux corrected transport (FCT) schemes: see Boris and Book (1973, 1976). Further work by Van Leer (1974, 1977a,b, 1979), Harten (1983, 1984), Sweby (1984), Roe (1985), Osher and Chakravarthy (1984) and many others has contributed to the development of present-day TVD schemes. In the next section we explain the fundamentals of the TVD methodology.

5.10.1 Generalisation of upwind-biased discretisation schemes

Consider the standard control volume discretisation of the one-dimensional convection–diffusion equation (5.3). Discretisation of the diffusion terms using the central differencing practice is standard and does not require any further consideration. It is the discretisation of the convective flux term that requires special attention. We assume that the flow is in the positive x-direction, so $u > 0$, and develop the TVD concept as a generalisation of upwind-biased expressions for the value of transported quantity ϕ at the east face of a one-dimensional control volume.

The standard upwind differencing (UD) scheme for the east face value of ϕ_e gives

$$\phi_e = \phi_P \tag{5.64}$$

A linear upwind differencing (LUD) scheme, which involves two upstream values, yields the following expression for ϕ_e:

$$\begin{aligned}\phi_e &= \phi_P + \frac{(\phi_P - \phi_W)}{\delta x}\frac{\delta x}{2} \\ &= \phi_P + \frac{1}{2}(\phi_P - \phi_W)\end{aligned} \tag{5.65}$$

This can be thought of as a second-order extension of the original UD estimate (5.64) of ϕ_e with a correction based on an upwind-biased estimate $(\phi_P - \phi_W)/\delta x$ of the gradient of ϕ multiplied by the distance $\delta x/2$ between node P and the east face. Another way of looking at this is to recall that our aim is to construct expressions for convective flux $F_e\phi_e$. Hence, for positive flow direction, the convective flux discretisation by means of the LUD scheme can be thought of as the sum of the basic UD convective flux $F_e\phi_P$ plus an additional flux contribution $F_e(\phi_P - \phi_W)/2$ to improve the order of accuracy.

The QUICK scheme (5.47) can be similarly rearranged in the form of the UD estimate plus a correction:

$$\phi_e = \phi_P + \frac{1}{8}[3\phi_E - 2\phi_P - \phi_W] \tag{5.66}$$

The central differencing (CD) scheme can be written as follows:

$$\begin{aligned}\phi_e &= \frac{(\phi_P + \phi_E)}{2} \\ &= \phi_P + \frac{1}{2}(\phi_E - \phi_P)\end{aligned} \tag{5.67}$$

We consider a generalization of the higher-order schemes in the following form:

$$\phi_e = \phi_P + \frac{1}{2}\psi(\phi_E - \phi_P) \tag{5.68}$$

where ψ is an appropriate function.

In choosing this form we express the convective flux at the east face as the sum of the flux $F_e\phi_P$ that is obtained when we use UD and an additional convective flux $F_e\psi(\phi_E - \phi_P)/2$. The extra contribution is connected in some way to the gradient of the transported quantity ϕ at the east face, as indicated by its central difference approximation $(\phi_E - \phi_P)$. It is easy to see that the central difference scheme (5.68) leads to function $\psi = 1$, but in sections 5.3–5.5 we have established that an additional convective flux based on this choice of ψ leads to wiggles in the solution if the grid is too coarse due to lack of transportiveness. The upwind scheme (5.64) corresponds to function $\psi = 0$, but this choice of ψ gave rise to false diffusion. Looking at the higher-order schemes we find that the LUD scheme (5.65) may be rewritten as

$$\phi_e = \phi_P + \frac{1}{2}\left(\frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right)(\phi_E - \phi_P) \tag{5.69}$$

Hence, for LUD the function is $\psi = (\phi_P - \phi_W)/(\phi_E - \phi_P)$.

After some algebra the QUICK expression (5.66) can be rewritten as

$$\phi_e = \phi_P + \frac{1}{2}\left[\left(3 + \frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right)\frac{1}{4}\right](\phi_E - \phi_P) \tag{5.70}$$

By comparing equation (5.70) with equation (5.68) it can be seen that the appropriate function for the QUICK scheme is

$$\psi = \left(3 + \frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right)\frac{1}{4}$$

Inspection of the forms (5.69) and (5.70) shows that the ratio of upwind-side gradient to downwind-side gradient $(\phi_P - \phi_W)/(\phi_E - \phi_P)$ determines the value of function ψ and the nature of the scheme. Therefore, we let

$$\boxed{\psi = \psi(r)} \tag{5.71}$$

with

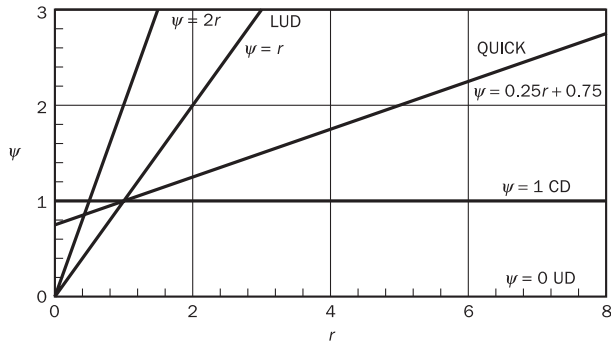
$$\boxed{r = \left(\frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right)}$$

The general form of the east face value ϕ_e within a discretisation scheme for convective flux may be written as

$$\boxed{\phi_e = \phi_P + \frac{1}{2}\psi(r)(\phi_E - \phi_P)} \tag{5.72}$$

For the UD scheme $\psi(r) = 0$
For the CD scheme $\psi(r) = 1$
For the LUD scheme $\psi(r) = r$
For the QUICK scheme $\psi(r) = (3 + r)/4$

Figure 5.21 shows the $\psi(r)$ vs. r relationships for these four schemes. This diagram is known as the $r-\psi$ diagram. All the above expressions assume that the flow direction is positive (i.e. from west to east). It can be shown that similar expressions exist for negative flow direction and r will still be the ratio of upwind-side gradient to downwind-side gradient.

Figure 5.21 The function ψ for various discretisation schemes

5.10.2 Total variation and TVD schemes

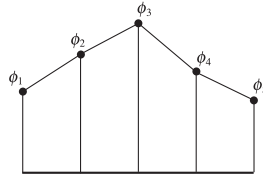
From our earlier discussion we know that the UD scheme is the most stable scheme and does not give any wiggles, whereas the CD and QUICK schemes have higher-order accuracy and give rise to wiggles under certain conditions. Our goal is to find a scheme with a higher-order of accuracy without wiggles. In our introduction to this topic it was noted that TVD schemes were

initially developed for time-dependent gas dynamics. In this context it has been established that the desirable property for a stable, non-oscillatory, higher-order scheme is **monotonicity preserving**. For a scheme to preserve monotonicity, (i) it must not create local extrema and (ii) the value of an existing local minimum must be non-decreasing and that of a local maximum must be non-increasing. In simple terms, monotonicity-preserving schemes do not create new undershoots and overshoots in the solution or accentuate existing extremes.

These properties of monotonicity-preserving schemes have implications for the so-called **total variation** of discretised solutions. Consider the discrete data set shown in Figure 5.22 (Lien and Leschziner, 1993). The total variation for this set of data is defined as

$$\begin{aligned} TV(\phi) &= |\phi_2 - \phi_1| + |\phi_3 - \phi_2| + |\phi_4 - \phi_3| + |\phi_5 - \phi_4| \\ &= |\phi_3 - \phi_1| + |\phi_5 - \phi_3| \end{aligned} \tag{5.73}$$

For monotonicity to be satisfied, this total variation must not increase (see Lien and Leschziner, 1993).

Figure 5.22 An example of a discrete data set for illustrating total variation

Monotonicity-preserving schemes have the property that the total variation of the discrete solution should diminish with time. Hence the term **total variation diminishing or TVD**. In the literature (Harten, 1983, 1984; Sweby, 1984) the total variation has been considered for *transient* one-dimensional transport equations. Total variation is therefore considered at every time step and a solution is said to be total variation diminishing (or TVD) if $TV(\phi^{n+1}) \le TV(\phi^n)$ where n and $n + 1$ refer to consecutive time steps. In the next sections we show how this property is also linked to desirable behaviour of discretisation schemes for *steady* convection–diffusion problems.

5.10.3 Criteria for TVD schemes

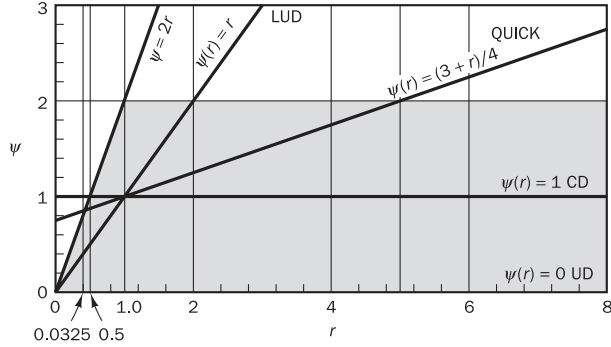
Sweby (1984) has given necessary and sufficient **conditions for a scheme to be TVD** in terms of the $r - \psi$ relationship:

- If $0 < r < 1$ the upper limit is $\psi(r) = 2r$, so for TVD schemes $\psi(r) \le 2r$
- If $r \ge 1$ the upper limit is $\psi(r) = 2$, so for TVD schemes $\psi(r) \le 2$

Figure 5.23 shows the shaded TVD region in a r–ψ diagram along with the r–ψ relationships for all the finite difference schemes we have discussed so far.

It can be seen that according to Sweby's criteria:

- the UD scheme is TVD
- the LUD scheme is not TVD for $r > 2.0$
- the CD scheme is not TVD for $r < 0.5$
- the QUICK scheme is not TVD for $r < 3/7$ and $r > 5$

Figure 5.23

Except for UD, all the above schemes are outside the TVD region for certain values of r. The idea of designing a TVD scheme is to introduce a modification to the above schemes so as to force the r–ψ relationship to remain within the shaded region for all possible values of r. This would imply that, in order to make the scheme TVD, we must constrain or limit the range of possible values of the additional convective flux $F_e\psi(r)(\phi_E - \phi_P)/2$, which was originally introduced to make the scheme higher-order. Hence, the function $\psi(r)$ is called a flux **limiter function**.

Sweby (1984) also introduced the following **requirement for second-order accuracy** in terms of the relationship $\psi = \psi(r)$:

- The flux limiter function of a second-order accurate scheme should pass through the point (1, 1) in the r–ψ diagram

Figure 5.23 confirms that the CD and QUICK schemes, which are both second-order accurate, satisfy this criterion, but the (first-order) UD scheme does not.

Sweby also showed that the **range of possible second-order schemes** is bounded by the central difference and linear upwind schemes:

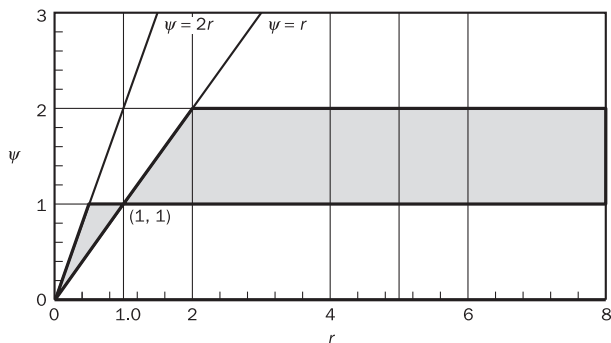
- If $0 < r < 1$ the lower limit is $\psi(r) = r$, the upper limit is $\psi(r) = 1$, so for TVD schemes $r \le \psi(r) \le 1$
- If $r \ge 1$ the lower limit is $\psi(r) = 1$, the upper limit is $\psi(r) = r$, so for TVD schemes $1 \le \psi(r) \le r$

The choice of $\psi(r)$ for a scheme dictates the order of the scheme and its boundedness properties. Any second-order limited scheme could be based on a limiter function which lies between $\psi(r) = r$ and $\psi(r) = 1$, goes through (1, 1) and stays below the upper limit. Any weighted average of the CD and LUD schemes that stays within the bounded region would, therefore, result in a second-order TVD scheme. Figure 5.24 shows the resulting shaded area for second-order TVD schemes.

Sweby finally introduced the **symmetry property** for limiter functions:

$$\frac{\psi(r)}{r} = \psi(1/r) \tag{5.74}$$

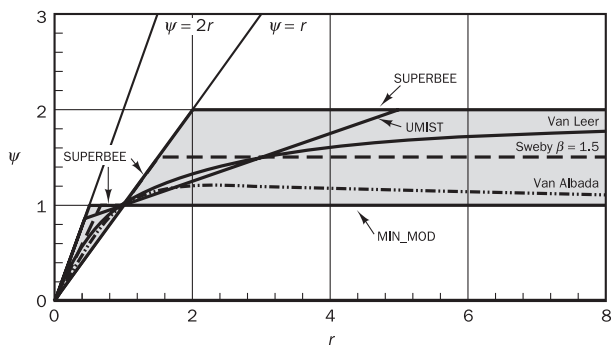
A limiter function that satisfies the symmetry property (5.74) ensures that backward- and forward-facing gradients are treated in the same fashion without the need for special coding.

Figure 5.24 Region for a second-order TVD scheme

5.10.4 Flux limiter functions

Over the years a number of limiters that satisfy Sweby's requirements have been developed and successfully used. Below we give some of the most popular limiter functions found in the literature:

Name	*Limiter function* $\psi(r)$	*Source*
Van Leer	$\dfrac{r+\lvert r\rvert}{1+r}$	Van Leer (1974)
Van Albada	$\dfrac{r+r^2}{1+r^2}$	Van Albada *et al.* (1982)
Min-Mod	$\psi(r)=\begin{cases}\min(r,1) & \text{if } r>0\\ 0 & \text{if } r\le 0\end{cases}$	Roe (1985)
SUPERBEE	$\max[0, \min(2r, 1), \min(r, 2)]$	Roe (1985)
Sweby	$\max[0, \min(\beta r, 1), \min(r, \beta)]$	Sweby (1984)
QUICK	$\max[0, \min(2r, (3+r)/4, 2)]$	Leonard (1988)
UMIST	$\max[0, \min(2r, (1+3r)/4, (3+r)/4, 2)]$	Lien and Leschziner (1993)

To compare the limiter functions we have plotted them all on the same r–ψ diagram in Figure 5.25. Separate figures for individual functions are shown in Appendix D.

Figure 5.25 All limiter functions in a r–ψ diagram

All the limiter functions stay inside the TVD region and pass through the point (1, 1) on the r–ψ diagram, so they all represent second-order accurate TVD discretisation schemes. Figure 5.25 shows that Van Leer and Van Albada's limiters are smooth functions, whereas all the others are piecewise linear expressions. The Min-Mod limiter function exactly traces the lower limit of the TVD region, whereas Roe's SUPERBEE scheme follows the upper limit. Sweby's expression is a generalisation of the Min-Mod and SUPERBEE limiters by means of a single parameter β. The limiter becomes the Min-Mod limiter when $\beta = 1$ and the SUPERBEE limiter of Roe when $\beta = 2$. To stay within the TVD region we only consider the range of values $1 \le \beta \le 2$. Figure 5.25 shows Sweby's limiter when $\beta = 1.5$. It is relatively easy to verify that Leonard's QUICK limiter function is the only one that is non-symmetric, whereas all the others are symmetric limiters. Lien and Leschziner's UMIST limiter function was designed as a symmetrical version of the QUICK limiter.

5.10.5 Implementation of TVD schemes

To demonstrate the most important aspects of the implementation of a TVD scheme we consider the now familiar one-dimensional convection–diffusion equation

$$\frac{\mathrm{d}}{\mathrm{d}x}(\rho u \phi) = \frac{\mathrm{d}}{\mathrm{d}x}\left[\Gamma \frac{\mathrm{d}\phi}{\mathrm{d}x}\right] \tag{5.3}$$

The diffusion term is discretised using central differencing as before, but the convective flux is now evaluated using a TVD scheme. In our usual notation the discretised form of the equation is as follows:

$$F_e\phi_e - F_w\phi_w = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{5.75}$$

For flow in the positive x-direction $u > 0$ and the values of ϕ_e and ϕ_w using a TVD scheme may be written as

$$\phi_e = \phi_P + \frac{1}{2}\psi(r_e)(\phi_E - \phi_P) \tag{5.76a}$$

$$\phi_w = \phi_W + \frac{1}{2}\psi(r_w)(\phi_P - \phi_W) \tag{5.76b}$$

$$\text{where } r_e = \left(\frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right) \text{ and } r_w = \left(\frac{\phi_W - \phi_{WW}}{\phi_P - \phi_W}\right)$$

Note that r for each face flux term is the local ratio of upstream gradient to downstream gradient. The limiter functions $\psi(r_e)$ and $\psi(r_w)$ can be any of the functions described above. Substitution of (5.76a) and (5.76b) into equation (5.75) gives

$$F_e\left[\phi_P + \frac{1}{2}\psi(r_e)(\phi_E - \phi_P)\right] - F_w\left[\phi_W + \frac{1}{2}\psi(r_w)(\phi_P - \phi_W)\right]$$
$$= D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)$$

This can be rearranged to yield

$$[D_e + F_e + D_w]\phi_P = [D_w + F_w]\phi_W + D_e\phi_E - F_e\left[\frac{1}{2}\psi(r_e)(\phi_E - \phi_P)\right] + F_w\left[\frac{1}{2}\psi(r_w)(\phi_P - \phi_W)\right] \quad (5.77)$$

This can be written as

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u^{DC} \quad (5.78a)$$

$$\text{where } a_W = D_w + F_w \quad (5.78b)$$

$$a_E = D_e \quad (5.78c)$$

$$a_P = a_W + a_E + (F_e - F_w) \quad (5.78d)$$

$$S_u^{DC} = -F_e\left[\frac{1}{2}\psi(r_e)(\phi_E - \phi_P)\right] + F_w\left[\frac{1}{2}\psi(r_w)(\phi_P - \phi_W)\right] \quad (5.78e)$$

It should be noted that the coefficients a_W, a_E and a_P are those of the UD scheme, which provides numerical stability to the TVD schemes. The contribution arising from the additional flux with the limiter function is introduced through the source term as a deferred correction S_u^{DC}. We have come across this practice before in section 5.9.3 when we discussed Hayase's implementation of the QUICK scheme. Deferred correction avoids the occurrence of stability problems due to negative coefficients in the discretised equation, whilst ensuring that the final converged solution has the desired TVD behaviour. As mentioned earlier, the above derivation is for the positive flow direction. To note the flow direction we use a superscript '+'. Therefore both r_e and r_w are replaced with r_w^+ and r_e^+. We rewrite the source term as

$$S_u^{DC} = -F_e\left[\frac{1}{2}\psi(r_e^+)(\phi_E - \phi_P)\right] + F_w\left[\frac{1}{2}\psi(r_w^+)(\phi_P - \phi_W)\right] \quad (5.79)$$

For $u < 0$, i.e. flow in the negative x-direction, the discretised form of the equation is as before:

$$F_e\phi_e - F_w\phi_w = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \quad (5.80)$$

Values of ϕ_e and ϕ_w using a TVD scheme are now

$$\phi_e = \phi_E + \frac{1}{2}\psi(r_e^-)(\phi_P - \phi_E) \quad (5.81a)$$

$$\phi_w = \phi_P + \frac{1}{2}\psi(r_w^-)(\phi_W - \phi_P) \quad (5.81b)$$

$$\text{where } r_e^- = \left(\frac{\phi_{EE} - \phi_E}{\phi_E - \phi_P}\right) \text{ and } r_w^- = \left(\frac{\phi_E - \phi_P}{\phi_P - \phi_W}\right)$$

Here we use the superscript '–' to indicate that the flow direction is in the negative x-direction. Note that r is still the local ratio of upstream gradient to downstream gradient. Substitution of (5.81a) and (5.81b) into equation (5.80) gives

$$F_e\left[\phi_E + \frac{1}{2}\psi(r_e^-)(\phi_P - \phi_E)\right] - F_w\left[\phi_P + \frac{1}{2}\psi(r_w^-)(\phi_W - \phi_P)\right]$$
$$= D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W)$$

The usual rearrangement yields

$$[D_e - F_w + D_w]\phi_P = D_w\phi_W + [D_e - F_e]\phi_E$$
$$+ F_e\left[\frac{1}{2}\psi(r_e^-)(\phi_E - \phi_P)\right] - F_w\left[\frac{1}{2}\psi(r_w^-)(\phi_P - \phi_W)\right] \tag{5.82}$$

This can be written as

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u^{DC} \tag{5.83a}$$
$$\text{where } a_W = D_w \tag{5.83b}$$
$$a_E = D_e - F_e \tag{5.83c}$$
$$a_P = a_W + a_E + (F_e - F_w) \tag{5.83d}$$
$$S_u^{DC} = F_e\left[\frac{1}{2}\psi(r_e^-)(\phi_E - \phi_P)\right] - F_w\left[\frac{1}{2}\psi(r_w^-)(\phi_P - \phi_W)\right] \tag{5.83e}$$

Again the expressions for the main coefficients are the same as for the UD scheme. We note that F_w and F_e are negative when the flow is in the negative x-direction, so coefficients a_W, a_E and a_P will always be positive. Combining expressions (5.78a–e) and (5.83a–e) we obtain a set of expressions valid for both positive and negative flow directions. Thus the **TVD scheme for one-dimensional convection–diffusion problems** may be written as

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u^{DC}} \tag{5.84}$$

with central coefficient

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

The neighbour coefficients and deferred correction source term of TVD schemes are as follows:

	TVD neighbour coefficients
a_W	$D_w + \max(F_w, 0)$
a_E	$D_e + \max(-F_e, 0)$
	TVD deferred correction source term
S_u^{DC}	$\frac{1}{2}F_e[(1-\alpha_e)\psi(r_e^-) - \alpha_e \,.\, \psi(r_e^+)](\phi_E - \phi_P) + \frac{1}{2}F_w[\alpha_w \,.\, \psi(r_w^+) - (1-\alpha_w)\psi(r_w^-)](\phi_P - \phi_W)$

where

$$\alpha_w = 1 \text{ for } F_w > 0 \text{ and } \alpha_e = 1 \text{ for } F_e > 0$$
$$\alpha_w = 0 \text{ for } F_w < 0 \text{ and } \alpha_e = 0 \text{ for } F_e < 0$$

Treatment at the boundaries

At inlet/outlet boundaries it is necessary to generate upstream/downstream values to evaluate the values of r. These can be obtained using the extrapolated mirror node practice that was demonstrated for the QUICK scheme in Example 5.4 (see section 5.9.1).

Consider an inlet with given boundary value $\phi = \phi_A$ and convective mass flux per unit area: $F = F_A$. The TVD discretised equation is

$$F_e\left[\phi_P + \frac{1}{2}\psi(r_e)(\phi_E - \phi_P)\right] - F_A\phi_A = D_e(\phi_E - \phi_P) - D_A^*(\phi_P - \phi_A)$$

$$(D_e + F_e + D_A^*)\phi_P = D_e\phi_E + (D_A^* + F_A)\phi_A - F_e\frac{1}{2}\psi(r_e)(\phi_E - \phi_P)$$

with $D_A^* = \Gamma/\delta x$

The problem is to find

$$r_e = \left(\frac{\phi_P - \phi_W}{\phi_E - \phi_P}\right)$$

for the deferred correction term. The gradient ratio contains a missing nodal value $\phi = \phi_W$.

Leonard mirror node extrapolation gives

$$\phi_o = 2\phi_A - \phi_P \quad \text{so} \quad r_e = \left(\frac{\phi_P - \phi_o}{\phi_E - \phi_P}\right) = \frac{2(\phi_P - \phi_A)}{\phi_E - \phi_P}$$

A further discussion on boundary conditions for higher-order schemes is available in Leonard (1988).

Extension to two and three dimensions

Extension of the TVD expressions to two dimensions is straightforward. The discretised equation using a **TVD scheme in a two-dimensional Cartesian grid** arrangement is given by

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + S_u^{DC}} \tag{5.85}$$

with central coefficient

$$\boxed{a_P = a_W + a_E + a_S + a_N + (F_e - F_w) + (F_n - F_s)}$$

The neighbour coefficients and deferred correction source term of TVD schemes are as follows:

	TVD neighbour coefficients
a_W	$D_w + \max(F_w, 0)$
a_E	$D_e + \max(-F_e, 0)$
a_S	$D_s + \max(F_s, 0)$
a_N	$D_n + \max(-F_n, 0)$
	TVD deferred correction source term
S_u^{DC}	$\frac{1}{2}F_e[(1-\alpha_e)\psi(r_e^-) - \alpha_e \,.\, \psi(r_e^+)](\phi_E - \phi_P)$ $+ \frac{1}{2}F_w[\alpha_w \,.\, \psi(r_w^+) - (1-\alpha_w)\psi(r_w^-)](\phi_P - \phi_W)$ $+ \frac{1}{2}F_n[(1-\alpha_n)\psi(r_n^-) - \alpha_n \,.\, \psi(r_n^+)](\phi_N - \phi_P)$ $+ \frac{1}{2}F_s[\alpha_s \,.\, \psi(r_s^+) - (1-\alpha_s)\psi(r_s^-)](\phi_P - \phi_S)$

where

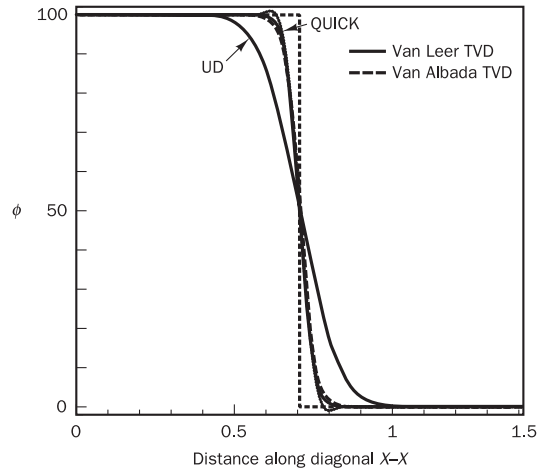
$$\alpha_w = 1 \text{ for } F_w > 0 \text{ and } \alpha_e = 1 \text{ for } F_e > 0$$
$$\alpha_w = 0 \text{ for } F_w < 0 \text{ and } \alpha_e = 0 \text{ for } F_e < 0$$
$$\alpha_s = 1 \text{ for } F_s > 0 \text{ and } \alpha_n = 1 \text{ for } F_n > 0$$
$$\alpha_s = 0 \text{ for } F_s < 0 \text{ and } \alpha_n = 0 \text{ for } F_n < 0$$

We note that the deferred correction source term now also includes terms related to south and north. The extension to three dimensions is straightforward.

5.10.6 Evaluation of TVD schemes

TVD schemes are generalisations of existing discretisation schemes, so they inherently satisfy all the necessary requirements of transportiveness, conservativeness and boundedness. In Figure 5.26 we compare the performance of two TVD schemes – Van Leer and Van Albada – with the UD and Leonard's QUICK schemes. The problem is the 2D source-free pure convection of a transported quantity ϕ with the flow at 45° to the lines of a 50 × 50 grid, which we considered previously in section 5.6.1. The exact solution to this problem is a step function at $x \approx 0.7$. It can be seen that TVD solutions show far less false diffusion than the UD scheme and are almost as close to the exact solution as the QUICK scheme. Moreover, they do not show any non-physical overshoots and undershoots. The two TVD solutions are quite close to each other, which is also a recurring feature in more broadly based performance comparisons in the literature.

Lien and Leschziner (1993) note that the more complex limiter functions take up more computer CPU time. Compared with an ordinary scheme, a calculation employing any TVD scheme would require more CPU time due to additional calculation overhead associated with evaluating the extra source terms (see section 5.10.5). The UMIST scheme, for example, was found to require 15% more CPU than the standard QUICK scheme (Lien and Leschziner, 1993). However, the advantage is that a TVD scheme guarantees wiggle-free solutions. There is no convincing argument in favour of any particular TVD scheme and the choice appears to be a matter of individual preference. The reader is also referred to the work of Darwish and Moukalled

Figure 5.26 Comparison of two TVD schemes: Van Leer and Van Albada with UD and QUICK

(2003), who describe the application of TVD schemes to unstructured mesh systems (see also Chapter 11).

5.11 Summary

We have discussed the problems of discretising the convection–diffusion equation under the assumption that the flow field is known. The crucial issue is the formulation of suitable expressions for the values of the transported property ϕ at cell faces when accounting for the convective contribution in the equation:

- All the finite volume schemes presented in this chapter describe the effects of simultaneous convection and diffusion by means of discretised equations whose coefficients are weighted combinations of the convective mass flux per unit area F and the diffusion conductance D.
- The discretised equations for a general internal node for the central, upwind and hybrid differencing and the power-law schemes of a one-dimensional convection–diffusion problem take the following form:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E} \tag{5.86}$$

  with

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

- The neighbour coefficients for these schemes are

Scheme	a_W	a_E
Central differencing	$D_w + F_w/2$	$D_e - F_e/2$
Upwind differencing	$D_w + \max(F_w, 0)$	$D_e + \max(0, -F_e)$
Hybrid differencing	$\max[F_w, (D_w + F_w/2), 0]$	$\max[-F_e, (D_e - F_e/2), 0]$
Power law	$D_w \max[0, (1 - 0.1\lvert Pe_w\rvert)^5] + \max(F_w, 0)$	$D_e \max[0, (1 - 0.1\lvert Pe_e\rvert)^5] + \max(-F_e, 0)$

- The boundary conditions enter the discretised equations via source terms. Their treatment is specific to each discretisation scheme.
- Discretisation schemes that possess conservativeness, boundedness and transportiveness give physically realistic results and stable iterative solutions:
  - The central differencing method is not suitable for general-purpose convection–diffusion problems because it lacks transportiveness and gives unrealistic solutions at large values of the cell Peclet number.
  - Upwind, hybrid and power-law differencing all possess conservativeness, boundedness and transportiveness and are highly stable, but suffer from false diffusion in multi-dimensional flows if the velocity vector is not parallel to one of the co-ordinate directions.
- The discretised equations of the standard QUICK method of Leonard (1979) have the following form for a general internal node point:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + a_{WW}\phi_{WW} + a_{EE}\phi_{EE}} \tag{5.87}$$

where

$$\boxed{a_P = a_W + a_E + a_{WW} + a_{EE} + (F_e - F_w)}$$

The neighbour coefficients of the standard QUICK scheme are

	Standard QUICK
a_W	$D_w + \frac{6}{8}\alpha_w F_w + \frac{1}{8}\alpha_e F_e + \frac{3}{8}(1-\alpha_w)F_w$
a_{WW}	$-\frac{1}{8}\alpha_w F_w$
a_E	$D_e - \frac{3}{8}\alpha_e F_e - \frac{6}{8}(1-\alpha_e)F_e - \frac{1}{8}(1-\alpha_w)F_w$
a_{EE}	$\frac{1}{8}(1-\alpha_e)F_e$

with

$\alpha_w = 1$ for $F_w > 0$ and $\alpha_e = 1$ for $F_e > 0$
$\alpha_w = 0$ for $F_w < 0$ and $\alpha_e = 0$ for $F_e < 0$

- Higher-order schemes, such as QUICK, can minimise false diffusion errors but are less computationally stable. This manifests itself as small over- and undershoots in the solution of some problems including those with large gradients of ϕ, which can potentially lead to non-physical behaviour, e.g. negative turbulence properties k and ε, in extreme cases. Nevertheless, if used with care and judgement the QUICK scheme can give very accurate solutions of convection–diffusion problems.

- The discretised equations of the TVD schemes have the following form for a general internal node point:

$$\boxed{a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u^{DC}} \tag{5.88}$$

where

$$\boxed{a_P = a_W + a_E + (F_e - F_w)}$$

The neighbour coefficients and deferred correction source term of TVD schemes are as follows:

	TVD neighbour coefficients
a_W	$a_W = D_w + \max(F_w, 0)$
a_E	$a_E = D_e + \max(-F_e, 0)$
	TVD deferred correction source term
S_u^{DC}	$\frac{1}{2}F_e[(1-\alpha_e)\psi(r_e^-) - \alpha_e \,.\, \psi(r_e^+)](\phi_E - \phi_P)$ $+ \frac{1}{2}F_w[\alpha_w \,.\, \psi(r_w^+) - (1-\alpha_w)\psi(r_w^-)](\phi_P - \phi_W)$

with

$\alpha_w = 1$ for $F_w > 0$ and $\alpha_e = 1$ for $F_e > 0$
$\alpha_w = 0$ for $F_w < 0$ and $\alpha_e = 0$ for $F_e < 0$

- The most frequently used limiter functions are

Name	***Limiter function ψ(r)***
Van Leer	$\dfrac{r+\lvert r\rvert}{1+r}$
Van Albada	$\dfrac{r+r^2}{1+r^2}$
Min-Mod	$\psi(r) = \begin{cases} \min(r, 1) & \text{if } r > 0 \\ 0 & \text{if } r \le 0 \end{cases}$
Roe's SUPERBEE	$\max[0, \min(2r, 1), \min(r, 2)]$
Sweby	$\max[0, \min(\beta r, 1), \min(r, \beta)]$
QUICK	$\max[0, \min(2r, (3+r)/4, 2)]$
UMIST	$\max[0, \min(2r, (1+3r)/4, (3+r)/4, 2)]$

- The performance of the limiter functions has been found to be fairly similar: all TVD discretisations based on the above limiter functions give second-order accurate solutions that are free from non-physical wiggles, so all are suitable for general-purpose CFD computations.

Chapter six Solution algorithms for pressure–velocity coupling in steady flows

6.1 Introduction

The convection of a scalar variable ϕ depends on the magnitude and direction of the local velocity field. To develop our methods in the previous chapter we assumed that the velocity field was somehow known. In general the velocity field is, however, not known and emerges as part of the overall solution process along with all other flow variables. In this chapter we look at the most popular strategies for computing the entire flow field.

Transport equations for each velocity component – momentum equations – can be derived from the general transport equation (2.39) by replacing the variable ϕ by u, v and w respectively. Every velocity component appears in each momentum equation, and the velocity field must also satisfy the continuity equation. This can be clearly shown by considering the equations governing a two-dimensional laminar steady flow:

x-momentum equation

$$\frac{\partial}{\partial x}(\rho uu) + \frac{\partial}{\partial y}(\rho vu) = \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right) - \frac{\partial p}{\partial x} + S_u \tag{6.1}$$

y-momentum equation

$$\frac{\partial}{\partial x}(\rho uv) + \frac{\partial}{\partial y}(\rho vv) = \frac{\partial}{\partial x}\left(\mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial y}\right) - \frac{\partial p}{\partial y} + S_v \tag{6.2}$$

continuity equation

$$\frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) = 0 \tag{6.3}$$

The pressure gradient term, which forms the main momentum source term in most flows of engineering importance, has been written separately to facilitate the discussion that follows.

The solution of equation set (6.1)–(6.3) presents us with two new problems:

- The convective terms of the momentum equations contain non-linear quantities: for example, the first term of equation (6.1) is the x-derivative of ρu^2.
- All three equations are intricately coupled because every velocity component appears in each momentum equation and in the continuity equation. The most complex issue to resolve is the role played by the

pressure. It appears in both momentum equations, but there is evidently no (transport or other) equation for the pressure.

If the pressure gradient is known, the process of obtaining discretised equations for velocities from the momentum equations is exactly the same as that for any other scalar, and the schemes explained in Chapter 5 are applicable. In general-purpose flow computations we also wish to calculate the pressure field as part of the solution, so its gradient is not normally known beforehand. If the flow is compressible the continuity equation may be used as the transport equation for density and, in addition to (6.1)–(6.3), the energy equation is the transport equation for temperature. The pressure may then be obtained from density and temperature by using the equation of state $p = p(\rho, T)$. However, if the flow is incompressible the density is constant and hence by definition not linked to the pressure. In this case coupling between pressure and velocity introduces a constraint in the solution of the flow field: if the correct pressure field is applied in the momentum equations the resulting velocity field should satisfy continuity.

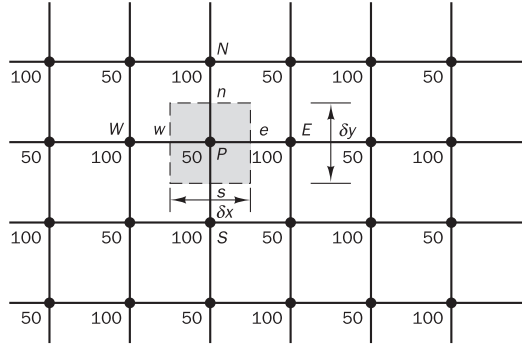
Both the problems associated with the non-linearities in the equation set and the pressure–velocity linkage can be resolved by adopting an iterative solution strategy such as the SIMPLE algorithm of Patankar and Spalding (1972). In this algorithm the convective fluxes per unit mass F through cell faces are evaluated from so-called guessed velocity components. Furthermore, a guessed pressure field is used to solve the momentum equations, and a pressure correction equation, deduced from the continuity equation, is solved to obtain a pressure correction field, which is in turn used to update the velocity and pressure fields. To start the iteration process we use initial guesses for the velocity and pressure fields. As the algorithm proceeds our aim must be progressively to improve these guessed fields. The process is iterated until convergence of the velocity and pressure fields. The main features of the SIMPLE algorithm and its more recent enhancements will be discussed in this chapter.

6.2 The staggered grid

The solution procedure for the transport of a general property ϕ developed in Chapter 5 will, of course, be enlisted to solve the momentum equations. Matters are, however, not completely straightforward since there are problems associated with the pressure source terms of the momentum equations that need special treatment.

The finite volume method starts, as always, with the discretisation of the flow domain and of the relevant transport equations (6.1)–(6.3). First we need to decide where to store the velocities. It seems logical to define these at the same locations as the scalar variables such as pressure, temperature etc. However, if velocities and pressures are both defined at the nodes of an ordinary control volume a highly non-uniform pressure field can act like a uniform field in the discretised momentum equations. This can be demonstrated with the simple two-dimensional situation shown in Figure 6.1, where a uniform grid is used for simplicity. Let us assume that we have somehow obtained a highly irregular 'checker-board' pressure field with values as shown in Figure 6.1.

If pressures at e and w are obtained by linear interpolation the pressure gradient term $\partial p/\partial x$ in the u-momentum equation is given by

Figure 6.1 A 'checker-board' pressure field

$$\frac{\partial p}{\partial x} = \frac{p_e - p_w}{\delta x} = \frac{\left(\dfrac{p_E + p_P}{2}\right) - \left(\dfrac{p_P + p_W}{2}\right)}{\delta x}$$

$$= \frac{p_E - p_W}{2\delta x} \tag{6.4}$$

Similarly, the pressure gradient $\partial p/\partial y$ for the v-momentum equation is evaluated as

$$\frac{\partial p}{\partial y} = \frac{p_N - p_S}{2\delta y} \tag{6.5}$$

The pressure at the central node (P) does not appear in (6.4) and (6.5). Substituting the appropriate values from the 'checker-board' pressure field in Figure 6.1 into (6.4)–(6.5) we find that all the discretised gradients are zero at all the nodal points even though the pressure field exhibits spatial oscillations in both directions. As a result, this pressure field would give the same (zero) momentum source in the discretised equations as a uniform pressure field. This behaviour is obviously non-physical.

It is clear that, if the velocities are defined at the scalar grid nodes, the influence of pressure is not properly represented in the discretised momentum equations. A remedy for this problem is to use a **staggered grid** for velocity components (Harlow and Welch, 1965). The idea is to evaluate scalar variables, such as pressure, density, temperature etc., at ordinary nodal points but to calculate velocity components on staggered grids centred around the cell faces. The arrangement for a two-dimensional flow calculation is shown in Figure 6.2.

The scalar variables, including pressure, are stored at the nodes marked (●). The velocities are defined at the (scalar) cell faces in between the nodes and are indicated by arrows. Horizontal (→) arrows indicate the locations for u-velocities and vertical (↑) ones denote those for v-velocity. In addition to the E, W, N, S notation Figure 6.2 also introduces a new system of notation based on a numbering of grid lines and cell faces. It will be explained and used later on in this chapter.

For the moment we continue to use the original E, W, N, S notation; the u-velocities are stored at scalar cell faces e and w and the v-velocities at faces n and s. In a three-dimensional flow the w-component is evaluated at cell faces t and b. We observe that the control volumes for u and v are

Figure 6.2

different from the scalar control volumes and from each other. The scalar control volumes are sometimes referred to as the pressure control volumes because, as we will see later, the discretised continuity equation is turned into a pressure correction equation, which is evaluated on scalar control volumes.

In the staggered grid arrangement, the pressure nodes coincide with the cell faces of the u-control volume. The pressure gradient term $\partial p/\partial x$ is given by

$$\frac{\partial p}{\partial x} = \frac{p_P - p_W}{\delta x_u} \tag{6.6}$$

where δx_u is the width of the u-control volume. Similarly $\partial p/\partial y$ for the v-control volume shown is given by

$$\frac{\partial p}{\partial y} = \frac{p_P - p_S}{\delta y_v} \tag{6.7}$$

where δy_v is the width of the v-control volume.

If we consider the ‘checker-board’ pressure field again, substitution of the appropriate nodal pressure values into equations (6.6) and (6.7) now yields very significant non-zero pressure gradient terms. The staggering of the velocity avoids the unrealistic behaviour of the discretised momentum equation for spatially oscillating pressures like the ‘checker-board’ field. A further advantage of the staggered grid arrangement is that it generates velocities at

exactly the locations where they are required for the scalar transport – convection–diffusion – computations. Hence, no interpolation is needed to calculate velocities at the scalar cell faces.

6.3 The momentum equations

As mentioned earlier, if the pressure field is known, the discretisation of velocity equations and the subsequent solution procedure is the same as that of a scalar equation. Since the velocity grid is staggered the new notation based on grid line and cell face numbering will be used. In Figure 6.2 the unbroken grid lines are numbered by means of capital letters. In the x-direction the numbering is . . . , $I-1$, I, $I+1$, . . . etc. and in the y-direction . . . , $J-1, J, J+1$, . . . etc. The dashed lines that construct the cell faces are denoted by lower case letters . . . , $i-1$, i, $i+1$, . . . and . . . , $j-1$, j, $j+1$, . . . in the x- and y-directions respectively.

A subscript system based on this numbering allows us to define the locations of grid nodes and cell faces with precision. Scalar nodes, located at the intersection of two grid lines, are identified by two capital letters: e.g. point P in Figure 6.2 is denoted by (I, J). The u-velocities are stored at the e- and w-cell faces of a scalar control volume. These are located at the intersection of a line defining a cell boundary and a grid line and are, therefore, defined by a combination of a lower case letter and a capital, e.g. the w-face of the cell around point P is identified by (i, J). For the same reasons the storage locations for the v-velocities are combinations of a capital and a lower case letter: e.g. the s-face is given by (I, j).

We may use forward or backward staggered velocity grids. The uniform grids in Figure 6.2 are backward staggered since the i-location for the u-velocity $u_{i,J}$ is at a distance of $-\frac{1}{2}\delta x_u$ from the scalar node (I, J). Likewise, the j-location for the v-velocity $v_{I,j}$ is $-\frac{1}{2}\delta y_v$ from node (I, J).

Expressed in the new co-ordinate system the discretised u-momentum equation for the velocity at location (i, J) is given by

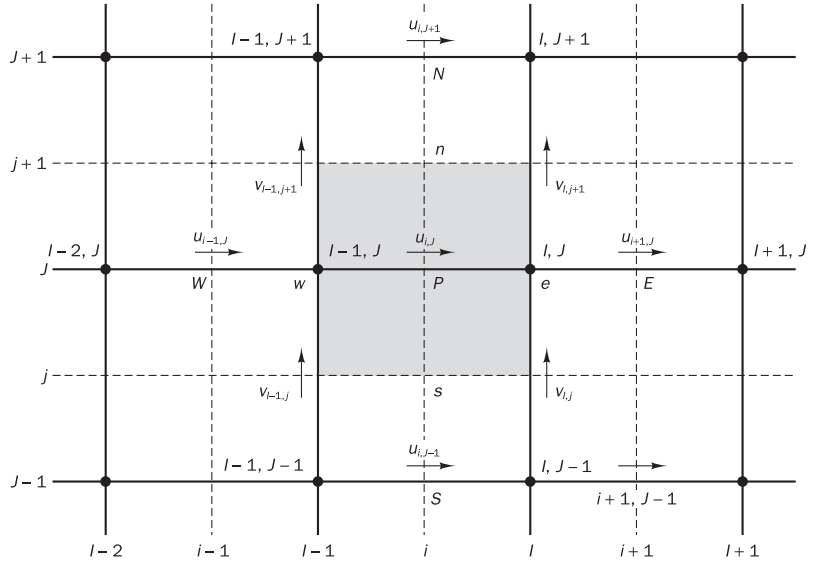
$$a_{i,J}u_{i,J} = \sum a_{nb}u_{nb} - \frac{p_{I,J} - p_{I-1,J}}{\delta x_u}\Delta V_u + \bar{S}\Delta V_u$$

or

$$a_{i,J}u_{i,J} = \sum a_{nb}u_{nb} + (p_{I-1,J} - p_{I,J})A_{i,J} + b_{i,J} \tag{6.8}$$

where ΔV_u is the volume of the u-cell, $b_{i,J} = \bar{S}\Delta V_u$ is the momentum source term, $A_{i,J}$ is the (east or west) cell face area of the u-control volume. The pressure gradient source term in (6.8) has been discretised by means of a linear interpolation between the pressure nodes on the u-control volume boundaries.

In the new numbering system the E, W, N and S neighbours involved in the summation $\sum a_{nb}u_{nb}$ are $(i-1, J)$, $(i+1, J)$, $(i, J-1)$ and $(i, J+1)$. Their locations and the prevailing velocities are shown in more detail in Figure 6.3. The values of coefficients $a_{i,J}$ and a_{nb} may be calculated with any of the differencing methods (upwind, hybrid, QUICK, TVD) suitable for convection–diffusion problems. The coefficients contain combinations of the convective flux per unit mass F and the diffusive conductance D at u-control volume cell faces. Applying the new notation system we give the values of F and D for each of the faces e, w, n and s of the u-control volume:

Figure 6.3 A u-control volume and its neighbouring velocity components

$$F_w = (\rho u)_w = \frac{F_{i,J} + F_{i-1,J}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J} + \rho_{I-1,J}}{2}\right)u_{i,J} + \left(\frac{\rho_{I-1,J} + \rho_{I-2,J}}{2}\right)u_{i-1,J}\right] \tag{6.9a}$$

$$F_e = (\rho u)_e = \frac{F_{i+1,J} + F_{i,J}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I+1,J} + \rho_{I,J}}{2}\right)u_{i+1,J} + \left(\frac{\rho_{I,J} + \rho_{I-1,J}}{2}\right)u_{i,J}\right] \tag{6.9b}$$

$$F_s = (\rho v)_s = \frac{F_{I,j} + F_{I-1,j}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J} + \rho_{I,J-1}}{2}\right)v_{I,j} + \left(\frac{\rho_{I-1,J} + \rho_{I-1,J-1}}{2}\right)v_{I-1,j}\right] \tag{6.9c}$$

$$F_n = (\rho v)_n = \frac{F_{I,j+1} + F_{I-1,j+1}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J+1} + \rho_{I,J}}{2}\right)v_{I,j+1} + \left(\frac{\rho_{I-1,J+1} + \rho_{I-1,J}}{2}\right)v_{I-1,j+1}\right] \tag{6.9d}$$

$$D_w = \frac{\Gamma_{I-1,J}}{x_i - x_{i-1}} \tag{6.9e}$$

$$D_e = \frac{\Gamma_{I,J}}{x_{i+1} - x_i} \tag{6.9f}$$

$$D_s = \frac{\Gamma_{I-1,J} + \Gamma_{I,J} + \Gamma_{I-1,J-1} + \Gamma_{I,J-1}}{4(y_J - y_{J-1})} \tag{6.9g}$$

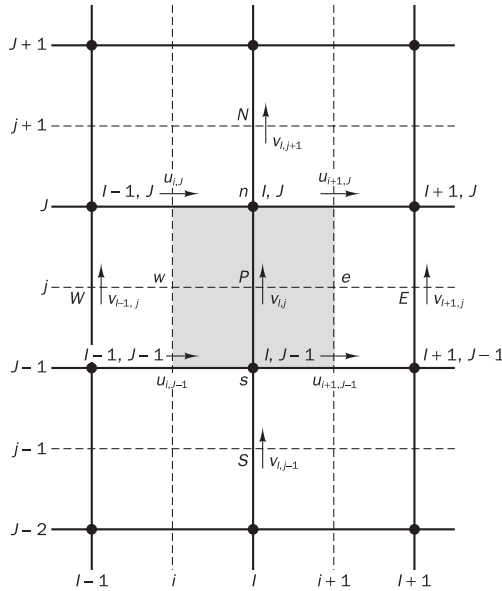
$$D_n = \frac{\Gamma_{I-1,J+1} + \Gamma_{I,J+1} + \Gamma_{I-1,J} + \Gamma_{I,J}}{4(y_{J+1} - y_J)} \tag{6.9h}$$

The formulae (6.9) show that where scalar variables or velocity components are not available at a u-control volume cell face, a suitable two- or four-point average is formed over the nearest points where values are available. During each iteration the u- and v-velocity components used to evaluate the above expressions are those obtained as the outcome of the previous iteration (or the initial guess in the first iteration). It should be noted that these **known** u- and v-values contribute to the coefficients a in equation (6.8). These are distinct from $u_{i,J}$ and u_{nb} in this equation, which denote the **unknown** scalars.

By analogy the v-momentum equation becomes

$$a_{I,j} v_{I,j} = \sum a_{nb} v_{nb} + (p_{I,J-1} - p_{I,J}) A_{I,j} + b_{I,j} \tag{6.10}$$

The neighbours involved in the summation $\sum a_{nb} v_{nb}$ and prevailing velocities are as shown in Figure 6.4.

Figure 6.4 A v-control volume and its neighbouring velocity components

Coefficients $a_{I,j}$ and a_{nb} again contain combinations of the convective flux per unit mass F and the diffusive conductance D at v-control volume cell faces. Their values are obtained by the same averaging procedure adopted for the u-control volume and are given below:

$$F_w = (\rho u)_w = \frac{F_{i,J} + F_{i,J-1}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J} + \rho_{I-1,J}}{2}\right) u_{i,J} + \left(\frac{\rho_{I-1,J-1} + \rho_{I,J-1}}{2}\right) u_{i,J-1}\right] \tag{6.11a}$$

$$F_e = (\rho u)_e = \frac{F_{i+1,J} + F_{i+1,J-1}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I+1,J} + \rho_{I,J}}{2}\right)u_{i+1,J} + \left(\frac{\rho_{I,J-1} + \rho_{I+1,J-1}}{2}\right)u_{i+1,J-1}\right] \quad (6.11b)$$

$$F_s = (\rho v)_s = \frac{F_{I,j-1} + F_{I,j}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J-1} + \rho_{I,J-2}}{2}\right)v_{I,j-1} + \left(\frac{\rho_{I,J} + \rho_{I,J-1}}{2}\right)v_{I,j}\right] \quad (6.11c)$$

$$F_n = (\rho v)_n = \frac{F_{I,j} + F_{I,j+1}}{2}$$

$$= \frac{1}{2}\left[\left(\frac{\rho_{I,J} + \rho_{I,J-1}}{2}\right)v_{I,j} + \left(\frac{\rho_{I,J+1} + \rho_{I,J}}{2}\right)v_{I,j+1}\right] \quad (6.11d)$$

$$D_w = \frac{\Gamma_{I-1,J-1} + \Gamma_{I,J-1} + \Gamma_{I-1,J} + \Gamma_{I,J}}{4(x_I - x_{I-1})} \quad (6.11e)$$

$$D_e = \frac{\Gamma_{I,J-1} + \Gamma_{I+1,J-1} + \Gamma_{I,J} + \Gamma_{I+1,J}}{4(x_{I+1} - x_I)} \quad (6.11f)$$

$$D_s = \frac{\Gamma_{I,J-1}}{y_j - y_{j-1}} \quad (6.11g)$$

$$D_n = \frac{\Gamma_{I,J}}{y_{j+1} - y_j} \quad (6.11h)$$

Again at each iteration level the values of F are computed using the u- and v- velocity components resulting from the previous iteration.

Given a pressure field p, discretised momentum equations of the form (6.8) and (6.10) can be written for each u- and v-control volume and then solved to obtain the velocity fields. If the pressure field is correct the resulting velocity field will satisfy continuity. As the pressure field is unknown, we need a method for calculating pressure.

6.4 The SIMPLE algorithm

The acronym SIMPLE stands for Semi-Implicit Method for Pressure-Linked Equations. The algorithm was originally put forward by Patankar and Spalding (1972) and is essentially a guess-and-correct procedure for the calculation of pressure on the staggered grid arrangement introduced above. The method is illustrated by considering the two-dimensional laminar steady flow equations in Cartesian co-ordinates.

To initiate the SIMPLE calculation process a pressure field p^* is guessed. **Discretised momentum equations** (6.8) and (6.10) **are solved** using the guessed pressure field to yield velocity components u^* and v^* as follows:

$$\boxed{a_{i,J}u^*_{i,J} = \sum a_{nb}u^*_{nb} + (p^*_{I-1,J} - p^*_{I,J})A_{i,J} + b_{i,J}} \quad (6.12)$$

$$\boxed{a_{I,j}v^*_{I,j} = \sum a_{nb}v^*_{nb} + (p^*_{I,J-1} - p^*_{I,J})A_{I,j} + b_{I,j}} \tag{6.13}$$

Now we define the correction p' as the difference between correct pressure field p and the guessed pressure field p^*, so that

$$\boxed{p = p^* + p'} \tag{6.14}$$

Similarly we define velocity corrections u' and v' to relate the correct velocities u and v to the guessed velocities u^* and v^*:

$$\boxed{u = u^* + u'} \tag{6.15}$$

$$\boxed{v = v^* + v'} \tag{6.16}$$

Substitution of the correct pressure field p into the momentum equations yields the correct velocity field (u, v). Discretised equations (6.8) and (6.10) link the correct velocity fields with the correct pressure field.

Subtraction of equations (6.12) and (6.13) from (6.8) and (6.10), respectively, gives

$$a_{i,J}(u_{i,J} - u^*_{i,J}) = \sum a_{nb}(u_{nb} - u^*_{nb}) + [(p_{I-1,J} - p^*_{I-1,J}) - (p_{I,J} - p^*_{I,J})]A_{i,J} \tag{6.17}$$

$$a_{I,j}(v_{I,j} - v^*_{I,j}) = \sum a_{nb}(v_{nb} - v^*_{nb}) + [(p_{I,J-1} - p^*_{I,J-1}) - (p_{I,J} - p^*_{I,J})]A_{I,j} \tag{6.18}$$

Using correction formulae (6.14)–(6.16) the equations (6.17)–(6.18) may be rewritten as follows:

$$a_{i,J}u'_{i,J} = \sum a_{nb}u'_{nb} + (p'_{I-1,J} - p'_{I,J})A_{i,J} \tag{6.19}$$

$$a_{I,j}v'_{I,j} = \sum a_{nb}v'_{nb} + (p'_{I,J-1} - p'_{I,J})A_{I,j} \tag{6.20}$$

At this point an approximation is introduced: $\sum a_{nb}u'_{nb}$ and $\sum a_{nb}v'_{nb}$ are dropped to simplify equations (6.19) and (6.20) for the velocity corrections. **Omission of these terms is the main approximation** of the SIMPLE algorithm. We obtain

$$u'_{i,J} = d_{i,J}(p'_{I-1,J} - p'_{I,J}) \tag{6.21}$$

$$v'_{I,j} = d_{I,j}(p'_{I,J-1} - p'_{I,J}) \tag{6.22}$$

$$\text{where } d_{i,J} = \frac{A_{i,J}}{a_{i,J}} \text{ and } d_{I,j} = \frac{A_{I,j}}{a_{I,j}} \tag{6.23}$$

Equations (6.21) and (6.22) describe the corrections to be applied to velocities through formulae (6.15) and (6.16), which gives

$$\boxed{u_{i,J} = u^*_{i,J} + d_{i,J}(p'_{I-1,J} - p'_{I,J})} \tag{6.24}$$

$$\boxed{v_{I,j} = v^*_{I,j} + d_{I,j}(p'_{I,J-1} - p'_{I,J})} \tag{6.25}$$

Similar expressions exist for $u_{i+1,J}$ and $v_{I,j+1}$:

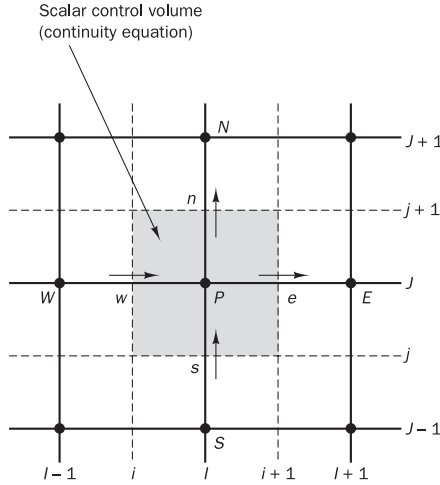
$$u_{i+1,J} = u^*_{i+1,J} + d_{i+1,J}(p'_{I,J} - p'_{I+1,J}) \tag{6.26}$$

$$v_{I,j+1} = v^*_{I,j+1} + d_{I,j+1}(p'_{I,J} - p'_{I,J+1}) \tag{6.27}$$

$$\text{where } d_{i+1,J} = \frac{A_{i+1,J}}{a_{i+1,J}} \text{ and } d_{I,j+1} = \frac{A_{I,j+1}}{a_{I,j+1}} \tag{6.28}$$

Thus far we have only considered the momentum equations but, as mentioned earlier, the velocity field is also subject to the constraint that it should satisfy continuity equation (6.3). Continuity is satisfied in discretised form for the scalar control volume shown in Figure 6.5:

$$[(\rho uA)_{i+1,J} - (\rho uA)_{i,J}] + [(\rho vA)_{I,j+1} - (\rho vA)_{I,j}] = 0 \tag{6.29}$$

Figure 6.5 The scalar control volume used for the discretisation of the continuity equation

Substitution of the corrected velocities of equations (6.24)–(6.27) into discretised continuity equation (6.29) gives

$$\begin{aligned}[]
[\rho_{i+1,J}A_{i+1,J}(u^*_{i+1,J} + d_{i+1,J}(p'_{I,J} - p'_{I+1,J})) - \rho_{i,J}A_{i,J}(u^*_{i,J} \\
+ d_{i,J}(p'_{I-1,J} - p'_{I,J}))] + [\rho_{I,j+1}A_{I,j+1}(v^*_{I,j+1} + d_{I,j+1}(p'_{I,J} - p'_{I,J+1})) \\
- \rho_{I,j}A_{I,j}(v^*_{I,j} + d_{I,j}(p'_{I,J-1} - p'_{I,J}))] = 0
\end{aligned} \tag{6.30}$$

This may be rearranged to give

$$\begin{aligned}[]
[(\rho dA)_{i+1,J} + (\rho dA)_{i,J} + (\rho dA)_{I,j+1} + (\rho dA)_{I,j}]p'_{I,J} = (\rho dA)_{i+1,J}p'_{I+1,J} \\
+ (\rho dA)_{i,J}p'_{I-1,J} + (\rho dA)_{I,j+1}p'_{I,J+1} + (\rho dA)_{I,j}p'_{I,J-1} \\
+ [(\rho u^*A)_{i,J} - (\rho u^*A)_{i+1,J} + (\rho v^*A)_{I,j} - (\rho v^*A)_{I,j+1}]
\end{aligned} \tag{6.31}$$

Identifying the coefficients of p', this may be written as

$$\boxed{a_{I,J}p'_{I,J} = a_{I+1,J}p'_{I+1,J} + a_{I-1,J}p'_{I-1,J} + a_{I,J+1}p'_{I,J+1} + a_{I,J-1}p'_{I,J-1} + b'_{I,J}} \tag{6.32}$$

where $a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1}$ and the coefficients are given below:

$a_{I+1,J}$	$a_{I-1,J}$	$a_{I,J+1}$	$a_{I,J-1}$	$b'_{I,J}$
$(\rho dA)_{i+1,J}$	$(\rho dA)_{i,J}$	$(\rho dA)_{I,j+1}$	$(\rho dA)_{I,j}$	$(\rho u^*A)_{i,J} - (\rho u^*A)_{i+1,J} + (\rho v^*A)_{I,j} - (\rho v^*A)_{I,j+1}$

Equation (6.32) represents the discretised continuity equation as an **equation for pressure correction p'**. The source term b' in the equation is the continuity imbalance arising from the incorrect velocity field u^*, v^*. By solving equation (6.32), the pressure correction field p' can be obtained at all points. Once the pressure correction field is known, the correct pressure field may be obtained using formula (6.14) and velocity components through correction formulae (6.24)–(6.27). The omission of terms such as $\sum a_{nb}u'_{nb}$ in the derivation does not affect the final solution because the pressure correction and velocity corrections will all be zero in a converged solution, giving $p^* = p$, $u^* = u$ and $v^* = v$.

The pressure correction equation is susceptible to divergence unless some **under-relaxation** is used during the iterative process, and new, improved, pressures p^{new} are obtained with

$$p^{new} = p^* + \alpha_p p' \tag{6.33}$$

where α_p is the pressure under-relaxation factor. If we select α_p equal to 1 the guessed pressure field p^* is corrected by p'. However, the corrections p', in particular when the guessed field p^* is far away from the final solution, is often too large for stable computations. A value of α_p equal to zero would apply no correction at all, which is also undesirable. Taking α_p between 0 and 1 allows us to add to guessed field p^* a fraction of the correction field p' that is large enough to move the iterative improvement process forward, but small enough to ensure stable computations.

The velocities are also under-relaxed. The iteratively improved velocity components u^{new} and v^{new} are obtained from

$$u^{new} = \alpha_u u + (1 - \alpha_u)u^{(n-1)} \tag{6.34}$$

$$v^{new} = \alpha_v v + (1 - \alpha_v)v^{(n-1)} \tag{6.35}$$

where α_u and α_v are the u- and v-velocity under-relaxation factors, u and v are the corrected velocity components without relaxation, and $u^{(n-1)}$ and $v^{(n-1)}$ represent their values obtained in the previous iteration. After some algebra it can be shown that with under-relaxation the discretised u-momentum equation takes the form

$$\frac{a_{i,J}}{\alpha_u}u_{i,J} = \sum a_{nb}u_{nb} + (p_{I-1,J} - p_{I,J})A_{i,J} + b_{i,J} + \left[(1 - \alpha_u)\frac{a_{i,J}}{\alpha_u}\right]u_{i,J}^{(n-1)} \tag{6.36}$$

and the discretised v-momentum equation

$$\frac{a_{I,j}}{\alpha_v}v_{I,j} = \sum a_{nb}v_{nb} + (p_{I,J-1} - p_{I,J})A_{I,j} + b_{I,j} + \left[(1 - \alpha_v)\frac{a_{I,j}}{\alpha_v}\right]v_{I,j}^{(n-1)} \tag{6.37}$$

The pressure correction equation is also affected by velocity under-relaxation, and it can be shown that d-terms of the pressure correction equation become

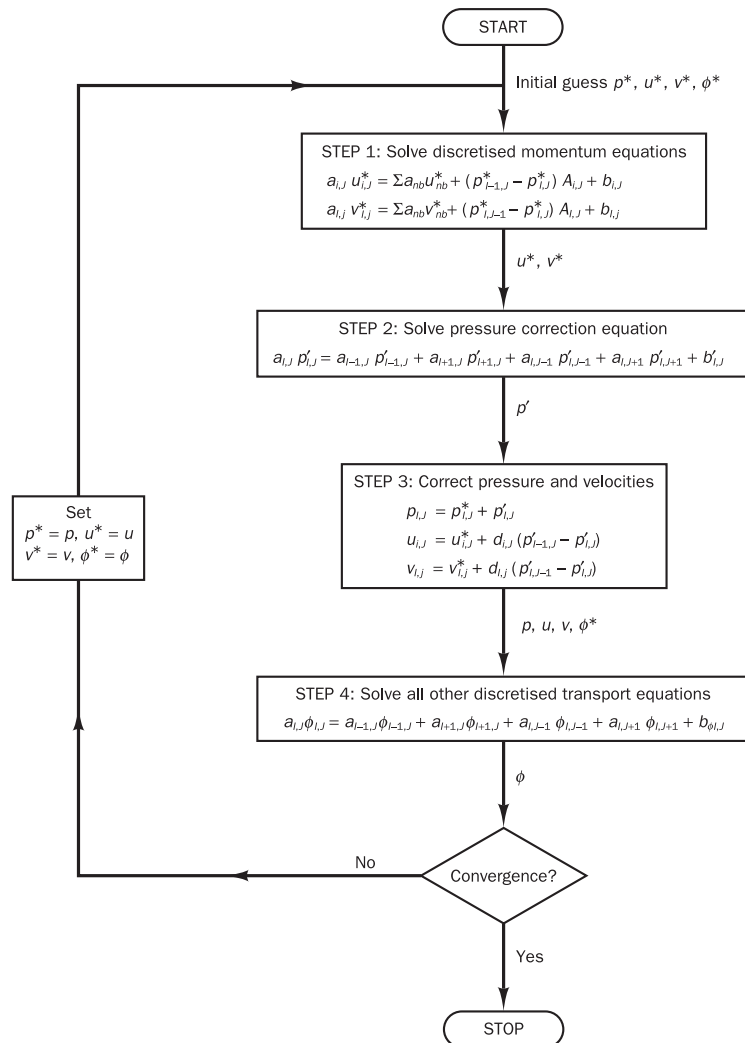
$$d_{i,J} = \frac{A_{i,J}\alpha_u}{a_{i,J}}, \quad d_{i+1,J} = \frac{A_{i+1,J}\alpha_u}{a_{i+1,J}}, \quad d_{I,j} = \frac{A_{I,j}\alpha_v}{a_{I,j}}, \quad d_{I,j+1} = \frac{A_{I,j+1}\alpha_v}{a_{I,j+1}}$$

Note that in these formulae $a_{i,J}$, $a_{i+1,J}$, $a_{I,j}$ and $a_{I,j+1}$ are the central coefficients of discretised velocity equations at positions (i, J), $(i + 1, J)$, (I, j) and $(I, j + 1)$ of a scalar cell centred around P.

A correct choice of under-relaxation factors α is essential for cost-effective simulations. Too large a value of α may lead to oscillatory or even divergent iterative solutions, and a value which is too small will cause extremely slow convergence. Unfortunately, the optimum values of under-relaxation factors are flow dependent and must be sought on a case-by-case basis. The use of under-relaxation will be discussed further in Chapters 7 and 8.

6.5 Assembly of a complete method

The SIMPLE algorithm gives a method of calculating pressure and velocities. The method is iterative, and when other scalars are coupled to the momentum equations the calculation needs to be done sequentially. The sequence of operations in a CFD procedure which employs the SIMPLE algorithm is given in Figure 6.6.

Figure 6.6 The SIMPLE algorithm

6.6 The SIMPLER algorithm

The SIMPLER (SIMPLE Revised) algorithm of Patankar (1980) is an improved version of SIMPLE. In this algorithm the discretised continuity equation (6.29) is used to derive a **discretised equation for pressure**, instead of a pressure correction equation as in SIMPLE. Thus the intermediate pressure field is obtained directly without the use of a correction. Velocities are, however, still obtained through the velocity corrections (6.24)–(6.27) of SIMPLE.

The discretised momentum equations (6.12)–(6.13) are rearranged as

$$u_{i,J} = \frac{\sum a_{nb}u_{nb} + b_{i,J}}{a_{i,J}} + \frac{A_{i,J}}{a_{i,J}}(p_{I-1,J} - p_{I,J}) \tag{6.38}$$

$$v_{I,j} = \frac{\sum a_{nb}v_{nb} + b_{I,j}}{a_{I,j}} + \frac{A_{I,j}}{a_{I,j}}(p_{I,J-1} - p_{I,J}) \tag{6.39}$$

In the SIMPLER algorithm pseudo-velocities $\hat{u}$ and $\hat{v}$ are now defined as follows:

$$\hat{u}_{i,J} = \frac{\sum a_{nb}u_{nb} + b_{i,J}}{a_{i,J}} \tag{6.40}$$

$$\hat{v}_{I,j} = \frac{\sum a_{nb}v_{nb} + b_{I,j}}{a_{I,j}} \tag{6.41}$$

Equations (6.38) and (6.39) can now be written as

$$u_{i,J} = \hat{u}_{i,J} + d_{i,J}(p_{I-1,J} - p_{I,J}) \tag{6.42}$$

$$v_{I,j} = \hat{v}_{I,j} + d_{I,j}(p_{I,J-1} - p_{I,J}) \tag{6.43}$$

The definition for d, introduced in the developments of section 6.4, is applied in (6.42)–(6.43). Substituting for $u_{i,J}$ and $v_{I,j}$ from these equations into the discretised continuity equation (6.29), using similar forms for $u_{i+1,J}$ and $v_{I,j+1}$, results in

$$\begin{aligned}&[\rho_{i+1,J}A_{i+1,J}(\hat{u}_{i+1,J} + d_{i+1,J}(p_{I,J} - p_{I+1,J})) - \rho_{i,J}A_{i,J}(\hat{u}_{i,J}\\ &\quad + d_{i,J}(p_{I-1,J} - p_{I,J}))] + [\rho_{I,j+1}A_{I,j+1}(\hat{v}_{I,j+1} + d_{I,j+1}(p_{I,J} - p_{I,J+1}))\\ &\quad - \rho_{I,j}A_{I,j}(\hat{v}_{I,j} + d_{I,j}(p_{I,J-1} - p_{I,J}))] = 0\end{aligned} \tag{6.44}$$

Equation (6.44) may be rearranged to give a discretised pressure equation

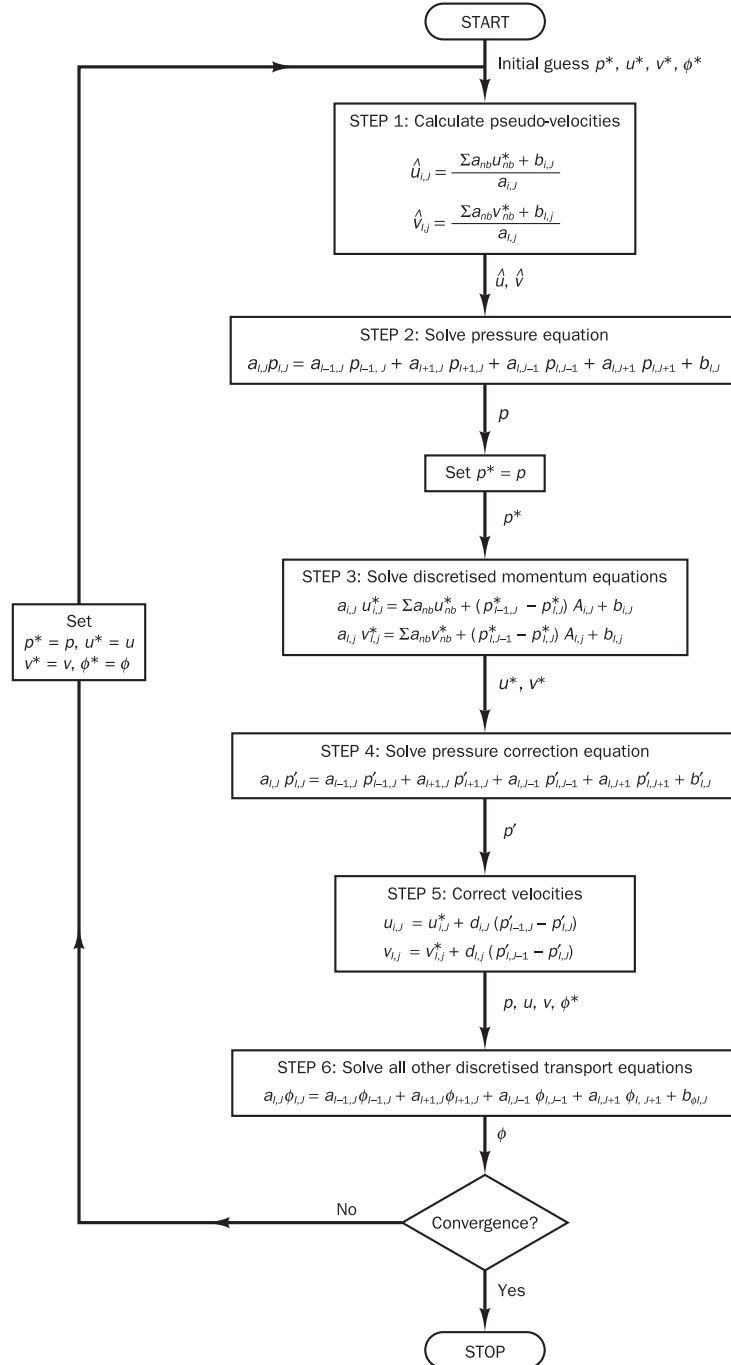
$$\boxed{a_{I,J}p_{I,J} = a_{I+1,J}p_{I+1,J} + a_{I-1,J}p_{I-1,J} + a_{I,J+1}p_{I,J+1} + a_{I,J-1}p_{I,J-1} + b_{I,J}} \tag{6.45}$$

where $a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1}$ and the coefficients are given below:

$a_{I+1,J}$	$a_{I-1,J}$	$a_{I,J+1}$	$a_{I,J-1}$	$b_{I,J}$
$(\rho dA)_{i+1,J}$	$(\rho dA)_{i,J}$	$(\rho dA)_{I,j+1}$	$(\rho dA)_{I,j}$	$(\rho\hat{u}A)_{i,J} - (\rho\hat{u}A)_{i+1,J} + (\rho\hat{v}A)_{I,j} - (\rho\hat{v}A)_{I,j+1}$

Note that the coefficients of equation (6.45) are the same as those in the discretised pressure correction equation (6.32), with the difference that the source term b is evaluated using the pseudo-velocities. Subsequently, the discretised momentum equations (6.12)–(6.13) are solved using the pressure

field obtained above. This yields the velocity components u^* and v^*. The velocity correction equations (6.24)–(6.27) are used in the SIMPLER algorithm to obtain corrected velocities. Therefore, the p'-equation (6.32) must also be solved to obtain the pressure corrections needed for the velocity corrections. The full sequence of operations is described in Figure 6.7.

Figure 6.7 The SIMPLER algorithm

6.7 The SIMPLEC algorithm

The SIMPLEC (SIMPLE-Consistent) algorithm of Van Doormal and Raithby (1984) follows the same steps as the SIMPLE algorithm, with the difference that the momentum equations are manipulated so that the SIMPLEC velocity correction equations omit terms that are less significant than those in SIMPLE.

The u-velocity correction equation of SIMPLEC is given by

$$u'_{i,J} = d_{i,J}(p'_{I-1,J} - p'_{I,J}) \tag{6.46}$$

$$\text{where } d_{i,J} = \frac{A_{i,J}}{a_{i,J} - \sum a_{nb}} \tag{6.47}$$

Similarly the modified v-velocity correction equation is

$$v'_{I,j} = d_{I,j}(p'_{I,J-1} - p'_{I,J}) \tag{6.48}$$

$$\text{where } d_{I,j} = \frac{A_{I,j}}{a_{I,j} - \sum a_{nb}} \tag{6.49}$$

The discretised pressure correction equation is the same as in SIMPLE, except that the d-terms are calculated from equations (6.47) and (6.49). The sequence of operations of SIMPLEC is identical to that of SIMPLE (see section 6.5).

6.8 The PISO algorithm

The PISO algorithm, which stands for Pressure Implicit with Splitting of Operators, of Issa (1986) is a pressure–velocity calculation procedure developed originally for non-iterative computation of unsteady compressible flows. It has been adapted successfully for the iterative solution of steady state problems. PISO involves one predictor step and two corrector steps and may be seen as an extension of SIMPLE, with **a further corrector step** to enhance it.

Predictor step

Discretised momentum equations (6.12)–(6.13) are solved with a guessed or intermediate pressure field p^* to give velocity components u^* and v^* using the same method as the SIMPLE algorithm.

Corrector step 1

The u^* and v^* fields will not satisfy continuity unless the pressure field p^* is correct. The first corrector step of SIMPLE is introduced to give a velocity field (u^{**}, v^{**}) which satisfies the discretised continuity equation. The resulting equations are the same as the velocity correction equations (6.21)–(6.22) of SIMPLE but, since there is a further correction step in the PISO algorithm, we use a slightly different notation:

$$p^{**} = p^* + p'$$
$$u^{**} = u^* + u'$$
$$v^{**} = v^* + v'$$

These formulae are used to define corrected velocities u^{**} and v^{**}:

$$u^{**}_{i,J} = u^*_{i,J} + d_{i,J}(p'_{I-1,J} - p'_{I,J}) \tag{6.50}$$

$$v^{**}_{I,j} = v^*_{I,j} + d_{I,j}(p'_{I,J-1} - p'_{I,J}) \tag{6.51}$$

As in the SIMPLE algorithm equations (6.50)–(6.51) are substituted into the discretised continuity equation (6.29) to yield pressure correction

equation (6.32) with its coefficients and source term. In the context of the PISO method equation (6.32) is called the first pressure correction equation. It is solved to yield the first pressure correction field p'. Once the pressure corrections are known, the velocity components u^{**} and v^{**} can be obtained through equations (6.50)–(6.51).

Corrector step 2

To enhance the SIMPLE procedure PISO performs a second corrector step. The discretised momentum equations for u^{**} and v^{**} are

$$a_{i,J}u^{**}_{i,J} = \sum a_{nb}u^{*}_{nb} + (p^{**}_{I-1,J} - p^{**}_{I,J})A_{i,J} + b_{i,J} \qquad (6.12)$$

$$a_{I,j}v^{**}_{I,j} = \sum a_{nb}v^{*}_{nb} + (p^{**}_{I,J-1} - p^{**}_{I,J})A_{I,j} + b_{I,j} \qquad (6.13)$$

A twice-corrected velocity field (u^{***}, v^{***}) may be obtained by solving the momentum equations once more:

$$a_{i,J}u^{***}_{i,J} = \sum a_{nb}u^{**}_{nb} + (p^{***}_{I-1,J} - p^{***}_{I,J})A_{i,J} + b_{i,J} \qquad (6.52)$$

$$a_{I,j}v^{***}_{I,j} = \sum a_{nb}v^{**}_{nb} + (p^{***}_{I,J-1} - p^{***}_{I,J})A_{I,j} + b_{I,j} \qquad (6.53)$$

Note that the summation terms are evaluated using the velocities u^{**} and v^{**} calculated in the previous step.

Subtraction of equation (6.12) from (6.52) and (6.13) from (6.53) gives

$$\boxed{u^{***}_{i,J} = u^{**}_{i,J} + \frac{\sum a_{nb}(u^{**}_{nb} - u^{*}_{nb})}{a_{i,J}} + d_{i,J}(p''_{I-1,J} - p''_{I,J})} \qquad (6.54)$$

$$\boxed{v^{***}_{I,J} = v^{**}_{I,j} + \frac{\sum a_{nb}(v^{**}_{nb} - v^{*}_{nb})}{a_{I,j}} + d_{I,j}(p''_{I,J-1} - p''_{I,J})} \qquad (6.55)$$

where p'' is the second pressure correction so that p^{***} may be obtained by

$$p^{***} = p^{**} + p'' \qquad (6.56)$$

Substitution of u^{***} and v^{***} in the discretised continuity equation (6.29) yields a second pressure correction equation

$$\boxed{a_{I,J}p''_{I,J} = a_{I+1,J}p''_{I+1,J} + a_{I-1,J}p''_{I-1,J} + a_{I,J+1}p''_{I,J+1} + a_{I,J-1}p''_{I,J-1} + b''_{I,J}} \qquad (6.57)$$

with $a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1}$, and the neighbour coefficients are as follows:

$a_{I+1,J}$	$a_{I-1,J}$	$a_{I,J+1}$	$a_{I,J-1}$	$b''_{I,J}$
$(\rho dA)_{i+1,J}$	$(\rho dA)_{i,J}$	$(\rho dA)_{I,j+1}$	$(\rho dA)_{I,j}$	$\left[\left(\frac{\rho A}{a}\right)_{i,J}\sum a_{nb}(u^{**}_{nb} - u^{*}_{nb}) - \left(\frac{\rho A}{a}\right)_{i+1,J}\sum a_{nb}(u^{**}_{nb} - u^{*}_{nb}) + \left(\frac{\rho A}{a}\right)_{I,j}\sum a_{nb}(v^{**}_{nb} - v^{*}_{nb}) - \left(\frac{\rho A}{a}\right)_{I,j+1}\sum a_{nb}(v^{**}_{nb} - v^{*}_{nb})\right]$

In the derivation of (6.57) the source term

$$[(\rho Au^{**})_{i,J} - (\rho Au^{**})_{i+1,J} + (\rho Av^{**})_{I,j} - (\rho Av^{**})_{I,j+1}]$$

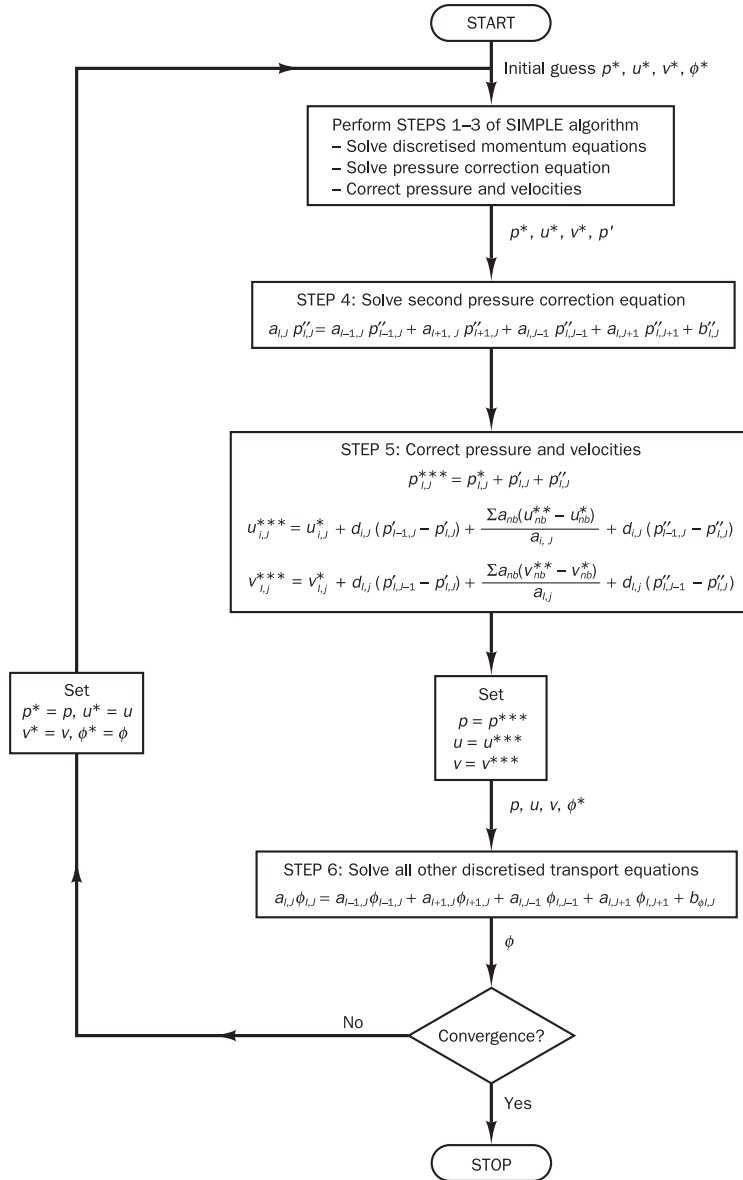
is zero since the velocity components u^{**} and v^{**} satisfy continuity.

Equation (6.57) is solved to obtain the second pressure correction field p'', and the twice-corrected pressure field is obtained from

$$\boxed{p^{***} = p^{**} + p'' = p^* + p' + p''} \quad (6.58)$$

Finally the twice-corrected velocity field is obtained from equations (6.54)–(6.55).

In the non-iterative calculation of unsteady flows the pressure field p^{***} and the velocity field u^{***} and v^{***} are considered to be the correct u, v and p. The sequence of operations for an iterative steady state PISO calculation is given in Figure 6.8.

Figure 6.8 The PISO algorithm

The PISO algorithm solves the pressure correction equation twice so the method requires additional storage for calculating the source term of the second pressure correction equation. As before, under-relaxation is required with the above procedure to stabilise the calculation process. Although this method implies a considerable increase in computational effort it has been found to be efficient and fast. For example, for a benchmark laminar backward-facing step problem Issa *et al.* (1986) reported a reduction of CPU time by a factor of 2 compared with standard SIMPLE.

The PISO algorithm presented above is the adapted, steady state version of an algorithm that was originally developed for non-iterative time-dependent calculations. The transient algorithm can also be applied to steady state calculations by starting with guessed initial conditions and solving as a transient problem for a long period of time until the steady state is achieved. This will be discussed in Chapter 8.

6.9 General comments on SIMPLE, SIMPLER, SIMPLEC and PISO

The SIMPLE algorithm is relatively straightforward and has been successfully implemented in numerous CFD procedures. The other variations of SIMPLE can produce savings in computational effort due to improved convergence. In SIMPLE, the pressure correction p' is satisfactory for correcting velocities but not so good for correcting pressure. Hence the improved procedure SIMPLER uses the pressure corrections to obtain velocity corrections only. A separate, more effective, pressure equation is solved to yield the correct pressure field. Since no terms are omitted to derive the discretised pressure equation in SIMPLER, the resulting pressure field corresponds to the velocity field. Therefore, in SIMPLER the application of the correct velocity field results in the correct pressure field, whereas it does not in SIMPLE. Consequently, the method is highly effective in calculating the pressure field correctly. This has significant advantages when solving the momentum equations. Although the number of calculations involved in SIMPLER is about 30% larger than that for SIMPLE, the fast convergence rate reportedly reduces the computer time by 30–50% (Anderson *et al.*, 1984). Further details of SIMPLE and its variants may be found in Patankar (1980).

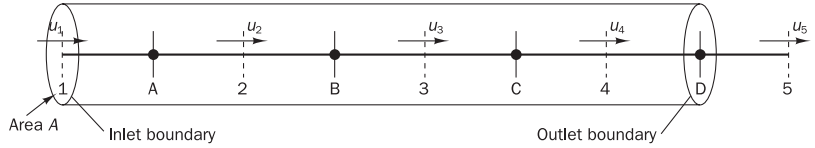
SIMPLEC and PISO have proved to be as efficient as SIMPLER in certain types of flows but it is not clear whether it can be categorically stated that they are better than SIMPLER. Comparisons have shown that the performance of each algorithm depends on the flow conditions, the degree of coupling between the momentum equation and scalar equations (in combusting flows, for example, due to the dependence of the local density on concentration and temperature), the amount of under-relaxation used, and sometimes even on the details of the numerical technique used for solving the algebraic equations. A comprehensive comparison of PISO, SIMPLER and SIMPLEC methods for a variety of steady flow problems by Jang *et al.* (1986) showed that, for problems in which momentum equations are not coupled to a scalar variable, PISO showed robust convergence behaviour and required less computational effort than SIMPLER and SIMPLEC. It was also observed that when the scalar variables were closely linked to velocities, PISO had no significant advantage over the other methods. Iterative methods using SIMPLER and SIMPLEC have robust convergence characteristics in strongly coupled problems, and it could not be ascertained which of SIMPLER or SIMPLEC was superior.

6.10 Worked examples of the SIMPLE algorithm

To illustrate the workings of the SIMPLE algorithm we give two detailed examples. To restrict the number of individual calculations we limit ourselves to one-dimensional flows as we have done in Chapters 4 and 5. In the first example we show how to update a velocity field in the case of a frictionless, incompressible flow through a duct of constant cross-sectional area. This problem has a trivial solution of constant velocity, but the example shows how an initial guess with varying velocities along the length of the duct is updated to satisfy mass conservation using the pressure correction equation. The second example looks at the frictionless, incompressible flow through a planar, converging nozzle. The nozzle shape cannot be accurately represented in the Cartesian x–y coordinate system that we have used until now. However, by making the assumption that the flow is unidirectional and all flow variables are uniformly distributed throughout every cross-section perpendicular to the flow direction, we can develop a set of one-dimensional governing equations for the problem. These exhibit the same pressure–velocity coupling issues as the two- and three-dimensional Navier–Stokes equations. Iterative solution of the discretised momentum equation and the pressure correction equation is needed to obtain the velocity and pressure field. We check the accuracy of the computed solution for our second example against the well-known Bernoulli equation.

Example 6.1

We consider the steady, one-dimensional flow of a constant-density fluid through a duct with constant cross-sectional area. We use the staggered grid shown in Figure 6.9, where the pressure p is evaluated at the main nodes I = A, B, C and D, whilst the velocity u is calculated at the backward staggered nodes i = 1, 2, 3 and 4.

Figure 6.9

As a starting point we assume that we have used a guessed pressure field p^* in the discretised momentum equation to obtain a guessed velocity field u^*. In this example we demonstrate the guess-and-correct procedure that forms the basis of the SIMPLE algorithm. Equation (6.32) is applied to generate pressure corrections p', which in turn yield velocity corrections u' by means of

$$u' = d(p'_I - p'_{I+1}) \tag{6.59}$$

and hence the corrected velocity field

$$u = u^* + u' \tag{6.60}$$

Problem data

The problem data are as follows:

- Density $\rho = 1.0$ kg/m^3 is constant.
- Duct area A is constant.

- Multiplier d in equation (6.59) is assumed to be constant; we take $d = 1.0$.
- Boundary conditions: $u_1 = 10$ m/s and $p_D = 0$ Pa.
- Initial guessed velocity field: say $u_2^* = 8.0$ m/s, $u_3^* = 11.0$ m/s and $u_4^* = 7.0$ m/s.

Use the SIMPLE algorithm and these problem data to calculate pressure corrections at nodes I = A to D and obtain the corrected velocity field at nodes $i = 2$ to 4. In this very straightforward problem with constant area and constant density it is easy to see that the velocity must be constant everywhere by continuity. Hence, we will be able to compare our computed solution against the exact solution $u_2 = u_3 = u_4 = 10$ m/s.

Solution

The pressure correction equation for this one-dimensional situation is equation (6.32):

$$a_P p'_P = a_W p'_W + a_E p'_E + b'$$

where $a_W = (\rho dA)_w$, $a_E = (\rho dA)_e$, $a_P = a_W + a_E$ and $b' = (\rho u^* A)_w - (\rho u^* A)_e$

Nodes B and C are internal nodes.

Node B

$$\begin{aligned}
a_W &= (\rho dA)_w = (\rho dA)_2 = 1.0 \times 1.0 \times A = 1.0A \\
a_E &= (\rho dA)_e = (\rho dA)_3 = 1.0 \times 1.0 \times A = 1.0A \\
a_P &= a_W + a_E = 1.0A + 1.0A = 2.0A \\
b' &= (\rho u^* A)_w - (\rho u^* A)_e = (\rho u^* A)_2 - (\rho u^* A)_3 \\
&= (1.0 \times 8. \times A) - (1.0 \times 11. \times A) = -3.0A
\end{aligned}$$

The discretised pressure correction equation at node B is

$$(2.0A)p'_B = (1.0A)p'_A + (1.0A)p'_C + (-3.0A)$$

The area A cancels on the left and right hand sides, which yields

$$2p'_B = p'_A + p'_C - 3$$

Node C

$$\begin{aligned}
a_W &= (\rho dA)_w = (\rho dA)_3 = 1.0 \times 1.0 \times A = 1.0A \\
a_E &= (\rho dA)_e = (\rho dA)_4 = 1.0 \times 1.0 \times A = 1.0A \\
a_P &= a_W + a_E = 1.0A + 1.0A = 2.0A \\
b' &= (\rho u^* A)_w - (\rho u^* A)_e = (\rho u^* A)_3 - (\rho u^* A)_4 \\
&= (1.0 \times 11. \times A) - (1.0 \times 7. \times A) = 4.0A
\end{aligned}$$

The discretised pressure correction equation at node C is

$$(2.0A)p'_C = (1.0A)p'_B + (1.0A)p'_D + (4.0A)$$
$$2p'_C = p'_B + p'_D + 4$$

Nodes A and D are boundary nodes.

Node A

We cut the link to the west boundary side by setting the relevant coefficient to zero and introduce the appropriate flux, in this case the mass flow rate into the control volume through the boundary side, as a source term b'.

$$
\begin{aligned}
a_W &= 0.0 \\
a_E &= (\rho dA)_e = (\rho dA)_2 = 1.0 \times 1.0 \times A = 1.0A \\
a_P &= a_W + a_E = 0.0 + 1.0A = 1.0A \\
b' &= (\rho u^* A)_w - (\rho u^* A)_e + (\rho uA)_{boundary} = -(\rho u^* A)_2 + (\rho uA)_1 = \\
&\qquad = -(1.0 \times 8. \times A) + (1.0 \times 10. \times A) \\
&\qquad = 2.0A
\end{aligned}
$$

Note that the given velocity at node 1 has been used in the calculation of the additional source contribution to b'. Using the above we obtain the discretised pressure correction equation at node A as

$$
\begin{aligned}
(1.0A)p'_A &= 0 + (1.0A)p'_B + (2.0A) \\
p'_A &= p'_B + 2.0
\end{aligned}
$$

Node D

The boundary condition at node D is fixed pressure $p_D = 0$. Since we know the pressure we do not need a pressure correction: hence at node D we have

$$p'_D = 0.$$

Thus we need to solve the following system of four equations for the four pressure corrections:

$$
\begin{aligned}
p'_A &= p'_B + 2 \\
2p'_B &= p'_A + p'_C - 3 \\
2p'_C &= p'_B + p'_D + 4 \\
p'_D &= 0
\end{aligned}
$$

We use $p'_D = 0$ directly in the pressure correction equation for node C, which becomes

$$2p'_C = p'_B + 4$$

This leaves a system of three equations with three unknowns. In matrix form the pressure correction equations are

$$
\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}
\begin{bmatrix} p'_A \\ p'_B \\ p'_C \end{bmatrix}
=
\begin{bmatrix} 2 \\ -3 \\ 4 \end{bmatrix}
$$

Solution of this set of equations gives

$$p'_A = 4.0,\ p'_B = 2.0 \text{ and } p'_C = 3.0 \text{ (with } p'_D = 0 \text{ as before).}$$

We obtain corrected velocities by combining (6.59) with (6.60):

$$u = u^* + d(p'_I - p'_{I+1})$$

Substitution of the problem data and the computed values for p' yields

Velocity node 2: $u_2 = u_2^* + d(p'_A - p'_B) = 8.0 + 1.0 \times [4.0 - 2.0] = 10.0$ m/s

Velocity node 3: $u_3 = u_3^* + d(p'_B - p'_C) = 11.0 + 1.0 \times [2.0 - 3.0] = 10.0$ m/s

Velocity node 4: $u_4 = u_4^* + d(p'_C - p'_D) = 7.0 + 1.0 \times [3.0 - 0.0] = 10.0$ m/s

This shows how the guess-and-correct procedure gives the exact velocity field in a single iteration for this very simple example. In more general flow

problems the pressure and velocity fields are coupled, so the pressure correction equation must be solved along with the discretised momentum equations. Furthermore, we note that the value of d in expression (6.59) for the velocity corrections was assumed to be constant. Normally, the value of d will vary from node to node and must be calculated with (6.23) and (6.28) using control volume face areas and central coefficient (a_P) values from the discretised momentum equations. This process will be illustrated in the next example.

Example 6.2

A planar two-dimensional nozzle is shown in Figure 6.10. The flow is steady and frictionless and the density of the fluid is constant.

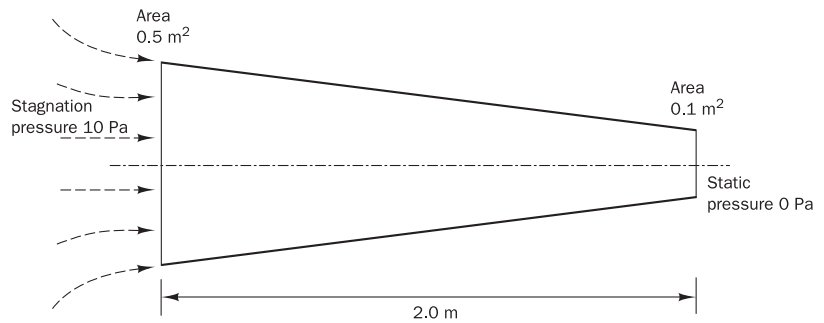
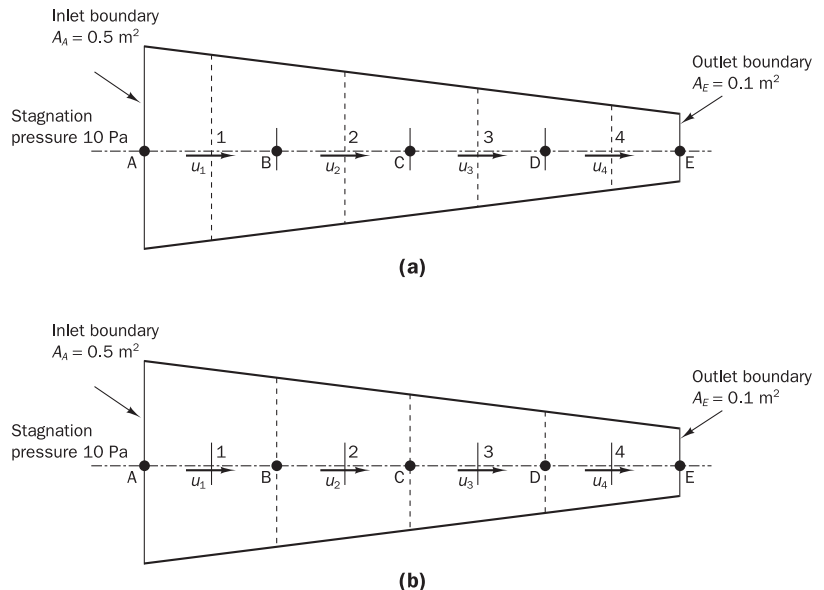
Figure 6.10 Geometry of planar 2D nozzle

Use the backward-staggered grid with five pressure nodes and four velocity nodes shown in Figures 6.11a–b. The stagnation pressure is given at the inlet and the static pressure is specified at the exit. Using the SIMPLE

Figure 6.11 (a) The grid for pressure control volumes; (b) the grid for velocity control volumes

algorithm write down the discretised momentum and pressure correction equations and solve for the unknown pressures at nodes I = B, C and D and velocities at nodes $i = 1, 2, 3$ and 4. Check whether the computed velocity field satisfies continuity and evaluate the error in the computed pressure and velocity fields by comparing with the exact solution.

Problem data

- The density of the fluid is 1.0 kg/m^3.
- Grid spacing: nozzle length $L = 2.00$ m. The grid is uniform so $\Delta x = L/4 = 2.00/4 = 0.5$ m.
- Cross-sectional area at the inlet $A_A = 0.5$ m^2 and at the exit is $A_E = 0.1$ m^2. The area change is a linear function of distance from the nozzle inlet. The table below gives the cross-sectional areas at all velocity and pressure nodes.
- Boundary conditions: at inlet we assume that the flow entering the nozzle is drawn from a large plenum chamber; the fluid has zero momentum and the stagnation pressure at inlet $p_0 = 10$ Pa. The static pressure at exit $p_E = 0$ Pa.
- Initial velocity field: to generate an initial velocity field for this problem we guess a mass flow rate say $\dot{m} = 1.0$ kg/s and use $u = \dot{m}/(\rho A)$ along with the cross-sectional areas at velocity nodes to compute the initial velocity field:

$$u_1 = \dot{m}/(\rho A_1) = 1.0/(1.0 \times 0.45) = 2.22222 \text{ m/s}$$
$$u_2 = \dot{m}/(\rho A_2) = 1.0/(1.0 \times 0.35) = 2.85714 \text{ m/s}$$
$$u_3 = \dot{m}/(\rho A_3) = 1.0/(1.0 \times 0.25) = 4.00000 \text{ m/s}$$
$$u_4 = \dot{m}/(\rho A_4) = 1.0/(1.0 \times 0.15) = 6.66666 \text{ m/s}$$

  N.B. five decimal places are shown throughout this example; the calculations have been performed with double precision accuracy.
- Initial pressure field: to generate a starting field of guessed pressures we assume a linear pressure variation between nodes A and E. Thus, $p_A^* = p_0 = 10.0$ Pa, $p_B^* = 7.5$ Pa, $p_C^* = 5.0$ Pa, $p_D^* = 2.5$ Pa and $p_E = 0.0$ Pa (given boundary condition).

The exact solution to this steady, one-dimensional, incompressible, frictionless flow problem can be obtained using Bernoulli's equation: $p_0 = p_N + \frac{1}{2}\rho u_N^2 = p_N + \frac{1}{2}\rho \dot{m}^2/(\rho A_N)^2$. From the problem data we have $p_0 = 10$ Pa, $\rho = 1$ kg/m^3 and N = E, so $A_N = A_E = 0.1$ m^2, which yields $\dot{m} = 0.44721$ kg/s. The nodal pressures and velocities are given in the table below.

Nozzle geometry and exact flow field using Bernoulli's equation

Node	*A (m^2)*	*p (Pa)*	*Node*	*A (m^2)*	*u (m/s)*
A	0.5	9.60000	1	0.45	0.99381
B	0.4	9.37500	2	0.35	1.27775
C	0.3	8.88889	3	0.25	1.78885
D	0.2	7.50000	4	0.15	2.98142
E	0.1	0			

Solution

The governing equations for steady, one-dimensional, incompressible, frictionless equations through the planar nozzle are as follows:

$$\text{Mass conservation:} \qquad \frac{d}{dx}(\rho A u) = 0 \tag{6.61}$$

$$\text{Momentum conservation:} \qquad \rho u A \frac{du}{dx} = -A\frac{dp}{dx} \tag{6.62}$$

These equations are familiar from introductory fluid mechanics texts. A derivation has also been given in Appendix E.

Discretised u-momentum equation

The discretised form of momentum equation (6.62) is

$$(\rho u A)_e u_e - (\rho u A)_w u_w = \frac{\Delta p}{\Delta x}\Delta V$$

where the right hand side represents the pressure gradient integrated over the control volume ΔV and $\Delta p = p_w - p_e$.

In standard notation the discretised momentum equation for this one-dimensional problem can be written as

$$a_P u_P^* = a_W u_W^* + a_E u_E^* + S_u$$

If we use the **upwind differencing scheme** the coefficients may be obtained from (see Section 5.6)

$$a_W = D_w + \max(F_w, 0)$$
$$a_E = D_e + \max(0, -F_e)$$
$$a_P = a_W + a_E + (F_e - F_w)$$

The flow is frictionless so there is no viscous diffusion term in the governing equation, and hence $D_w = D_e = 0$. F_w and F_e are mass flow rates through the west and east faces of the u-control volume. We compute the face velocities needed for F_w and F_e from averages of velocity values at nodes straddling the face and use the correct values of the west and east face area given in the above table. At the start of the calculations we use the initial velocity field generated from the guessed mass flow rate. For subsequent iterations we use the corrected velocity obtained after solving the pressure correction equation.

The source term S_u contains the pressure gradient integrated over the control volume:

$$S_u = \frac{\Delta p}{\Delta x} \times \Delta V = \frac{\Delta p}{\Delta x} \times A_{av}\Delta x = \Delta p \times \frac{1}{2}(A_w + A_e)$$

Since the geometry has a varying cross-sectional area we use an averaged area to calculate ΔV. At first glance this looks like a very crude approximation, but it is possible to show that the accuracy order of S_u is no worse than the upstream differencing used for the momentum flux terms.

In summary the coefficients of the discretised u-equations are given by

$$F_w = \rho A_w u_w \text{ and } F_e = \rho A_e u_e$$
$$a_W = F_w$$
$$a_E = 0$$

$$a_P = a_W + a_E + (F_e - F_w)$$
$$S_u = \Delta p \times \tfrac{1}{2}(A_w + A_e) = \Delta p \times A_P$$

The parameter d required in the pressure correction equations is calculated at this stage from

$$d = \frac{A_{av}}{a_P} = \frac{(A_w + A_e)}{2a_P}$$

Pressure correction equation

The discretised form of the continuity equation in this one-dimensional geometry is

$$(\rho uA)_e - (\rho uA)_w = 0$$

The corresponding pressure correction equation is

$$a_P p'_P = a_W p'_W + a_E p'_E + b'$$

where $a_W = (\rho dA)_w$, $a_E = (\rho dA)_e$
$b' = (F^*_w - F^*_e)$

Values of the parameter d come from discretised momentum equations (see above and Section 6.4).

In the SIMPLE algorithm the pressure corrections p' are used to compute the velocity corrections u' and the corrected pressure and velocity fields using

$$u' = d(p'_I - p'_{I+1})$$

$$p = p^* + p'$$

$$u = u^* + u'$$

Numerical values – momentum equation

First we consider the internal nodes 2 and 3.

- *Velocity node 2*

$$F_w = (\rho uA)_w = 1.0 \times [(u_1 + u_2)/2] \times 0.4$$
$$= 1.0 \times [(2.2222 + 2.8571)/2] \times 0.4 = 1.01587$$

$$F_e = (\rho uA)_e = 1.0 \times [(u_2 + u_3)/2] \times 0.3$$
$$= 1.0 \times [(2.8571 + 4.0)/2] \times 0.3 = 1.02857$$

$$a_W = F_w = 1.01587$$

$$a_E = 0$$

$$a_P = a_W + a_E + (F_e - F_w) = 1.01587 + 0 + (1.02857 - 1.01587)$$
$$= 1.02857$$

$$S_u = \Delta P \times A_2 = (p_B - p_C) \times A_2 = (7.5 - 5.0) \times 0.35 = 0.875$$

The discretised momentum equation at node 2 is

$$1.02857u_2 = 1.01587u_1 + 0.875$$

We also need to calculate the parameter d at this node for later use in the pressure correction equation:

$$d_2 = A_2/a_P = 0.35/1.02857 = 0.34027$$

- *Velocity node 3*

  We leave it as an exercise for the reader to check that application of the above procedure at the control volume around node 3 yields

  $$1.06666u_3 = 1.02857u_2 + 0.625$$

  and

  $$d_3 = A_3/a_P = 0.25/1.06666 = 0.23437$$

Next we come to momentum control volumes 1 and 4, which need special treatment because they both contain a boundary face.

- *Velocity node 1*

  The stagnation pressure $p_0 = 10$ Pa is given in a plenum chamber upstream from the inlet where the fluid is at rest. To carry out the calculations we need conditions at the actual inlet plane of the momentum control volume 1, which coincides with pressure node A. At this location the velocity is non-zero and the actual pressure is lower than the stagnation pressure due to acceleration of the flow as it enters the nozzle. We denote the (as yet unknown) velocity at A by u_A and use Bernoulli's equation to express the static pressure at A, which is needed in the source term S_u, in terms of p_0 and u_A:

  $$p_A = p_0 - \frac{1}{2}(\rho u_A^2) \tag{6.63}$$

  Next we write u_A in terms of the velocity u_1 using continuity:

  $$u_A = u_1 A_1/A_A \tag{6.64}$$

  Combining (6.63) and (6.64) yields

  $$p_A = p_0 - \frac{1}{2}\rho u_1^2\left(\frac{A_1}{A_A}\right)^2 \tag{6.65}$$

  Now we may write the discretised momentum equation for u-momentum control volume 1 using the upwind scheme:

  $$F_e u_1 - F_w u_A = (p_A - p_B) \times A_1 \tag{6.66}$$

  F_w is calculated using the estimate u_A from equation (6.64): i.e. $F_w = \rho u_A A_A = \rho u_1 A_1$.

  Substitution of expressions (6.64) and (6.65) into (6.66) gives

  $$F_e u_1 - F_w u_1 A_1/A_A = [(p_0 - \tfrac{1}{2}\rho u_1^2(A_1/A_A)^2) - p_B] \times A_1 \tag{6.67}$$

  Some rearrangement and placing all the terms involving pressures on the right and those involving velocities on the left hand side yields

  $$[F_e - F_w A_1/A_A + F_w \times \tfrac{1}{2}(A_1/A_A)^2]u_1 = (p_0 - p_B)A_1 \tag{6.68}$$

  Hence, the central coefficient a_P for this node is $a_P = F_e - F_w A_1/A_A + F_w \times \frac{1}{2}(A_1/A_A)^2$. The first two contributions on the right hand side of this formula come from the mass flux terms on the left hand side of discretised momentum equation (6.66). The third term is an extra contribution arising from our choice to specify the *stagnation* pressure at inlet (this extra term would be omitted if a value of the *static* pressure was specified at inlet instead).

Expression (6.68) can be used in that form, but in these calculations we have chosen to place the negative contribution to coefficient a_1 on the right hand side. Hence

$$[F_e + F_w \times \tfrac{1}{2}(A_1/A_A)^2]u_1 = (p_0 - p_B)A_1 + F_w A_1/A_A \times u_1^{old} \qquad (6.69)$$

where u_1^{old} is the nodal velocity at the previous iteration

This is termed the deferred correction approach and can be effective in stabilising the iterative process if the initial velocity field is based on a very poor guess (see also Chapter 5 – QUICK and TVD).

Now we calculate

$$u_A = u_1 A_1/A_A = 2.22222 \times 0.45/0.5 = 2.0$$

$$F_w = (\rho u A)_w = \rho u_A A_A = 1.0 \times 2.0 \times 0.5 = 1.0$$

The exit mass flux F_e is computed in the same way as for the internal nodes:

$$F_e = (\rho u A)_e = 1.0 \times [(u_1 + u_2)/2] \times 0.4$$
$$= 1.0 \times [(2.2222 + 2.8571)/2] \times 0.4 = 1.01587$$

$$a_W = 0$$

$$a_E = 0$$

$$a_P = F_e + F_w \times \tfrac{1}{2}(A_1/A_A)^2 = 1.01587 + 1.0 \times 0.5 \times (0.45/0.5)^2$$
$$= 1.42087$$

In the source term we apply $p_0 = 10$ Pa and the initial velocity $u_1^{old} = 2.22222$ m/s.

$$S_u = (p_0 - p_B)A_1 + F_w(A_1/A_A) \times u_1^{old}$$
$$= (10 - 7.5) \times 0.45 + 1.0 \times (0.45/0.5) \times 2.22222$$
$$= 3.125$$

The discretised momentum equation at node 1 is therefore

$$1.42087u_1 = 3.125$$

The parameter d at this node is

$$d_1 = A_1/a_P = 0.45/1.4209 = 0.31670$$

- *Velocity node 4*

$$F_w = (\rho u A)_w = 1.0 \times [(u_3 + u_4)/2] \times 0.2 = 1.06666$$

At the east boundary of momentum control volume 4 we have a fixed pressure, but we do not have two velocities that straddle the east boundary. To compute the mass flux across this boundary we impose continuity:

$$F_e = (\rho u A)_4$$

At the first iteration we can use the assumed mass flow rate, so we set $F_e = 1.0$ kg/s. Thus,

$$a_W = F_w = 1.06666$$
$$a_E = 0$$
$$a_P = a_W + a_E + (F_e - F_w) = 1.06666 + 0 + (1.0 - 1.06666) = 1.0$$

In the momentum source term we apply the given exit boundary pressure $p_E = 0$ Pa:

$$S_u = \Delta P \times A_{av} = (p_D - p_E) \times A_4 = (2.5 - 0.0) \times 0.15 = 0.375$$

The discretised momentum equation at node 4 is

$$1.0u_4 = 1.0666u_3 + 0.375$$

The parameter d at this node is

$$d_4 = A_4/a_P = 0.15/1.0 = 0.15$$

To summarise, the u-momentum equations using upwind differencing are as follows:

$$\begin{aligned}
1.42087u_1 &= 3.125 \\
1.02857u_2 &= 1.01587u_1 + 0.875 \\
1.06666u_3 &= 1.02857u_2 + 0.625 \\
1.00000u_4 &= 1.06666u_3 + 0.375
\end{aligned}$$

These equations can be solved by forward substitution starting at node 1. The solution is

u_1 m/s	u_2 m/s	u_3 m/s	u_4 m/s
2.19935	3.02289	3.50087	4.10926

These are the guessed velocities used in the SIMPLE pressure correction procedure. Therefore star (*) superscripts are used to refer to these u-values in the pressure correction calculations below.

The d values are as follows:

d_1	d_2	d_3	d_4
0.31670	0.34027	0.23437	0.15000

Numerical values – pressure correction equation

The internal nodes are B, C and D.

- *Pressure node B*

$$a_W = (\rho d A)_1 = 1.0 \times 0.3167 \times 0.45 = 0.14251$$

$$a_E = (\rho d A)_2 = 1.0 \times 0.34027 \times 0.35 = 0.11909$$

$$F_w^* = (\rho u^* A)_1 = 1.0 \times 2.199352 \times 0.45 = 0.98971$$

$$F_E^* = (\rho u^* A)_2 = 1.0 \times 3.022894 \times 0.35 = 1.05801$$

$$a_P = a_W + a_E = 0.14251 + 0.11909 = 0.26161$$

$$b' = F_w^* - F_e^* = 0.98971 - 1.05801 = -0.06830$$

The pressure correction equation at node B is

$$0.26161p'_B = 0.14251p'_A + 0.11909p'_C - 0.06830$$

- *Pressure nodes C and D*

We leave it for the reader to check that the corresponding pressure correction equations for nodes C and D are

$$0.17769p'_C = 0.11909p'_B + 0.058593p'_D + 0.18279$$
$$0.081093p'_D = 0.058593p'_C + 0.02249p'_E + 0.25882$$

Nodes are A and E are boundary nodes so they need special treatment.

- *Pressure nodes A and E*

  The pressure corrections are set to zero for both nodes:

$$p'_A = 0.0$$
$$p'_E = 0.0$$

  At node E this follows the practice of Example 6.1, because the static pressure is given at the nozzle exit. If the static pressure p_A at inlet was given this would also apply at node A without reservation. However, in this problem we are working with a given stagnation pressure, so we need to be careful. We note that p_A is fixed by Equation (6.65) if the stagnation pressure p_0 and velocity u_1 are known. At the stage in the SIMPLE algorithm where we start to solve the pressure correction equations, we have available the guessed velocity u_1^* as a result of solving the discretised momentum equation. Whilst it is true that this velocity is constantly updated as the iterations proceed, we may regard that at each iteration level the static pressure p_A is temporarily fixed by p_0 and the current value of u_1^*, thus justifying the use of $p'_A = 0.0$.

Substitution of $p'_A = 0.0$ and $p'_E = 0.0$ into the pressure correction equations for internal nodes B, C and D yields the following system of equations:

$$0.26161p'_B = 0.11909p'_C - 0.06830$$
$$0.17769p'_C = 0.11909p'_B + 0.058593p'_D + 0.18279$$
$$0.081093p'_D = 0.058593p'_C + 0.25882$$

These three equations can be solved to give the pressure correction at nodes B, C and D. The resulting solution is

p'_A	p'_B	p'_C	p'_D	p'_E
0.0	1.63935	4.17461	6.20805	0.0

Corrected nodal pressures are now calculated using these pressure corrections:

$$p_B = p_B^* + p'_B = 7.5 + 1.63935 = 9.13935$$
$$p_C = p_C^* + p'_C = 5.0 + 4.17461 = 9.17461$$
$$p_D = p_D^* + p'_D = 2.5 + 6.20805 = 8.70805$$

Corrected velocities at the end of the first iteration are

$$u_1 = u_1^* + d_1(p'_A - p'_B) = 2.19935 + 0.31670 \times [0.0 - 1.63935] = 1.68015 \text{ m/s}$$
$$u_2 = u_2^* + d_2(p'_B - p'_C) = 3.02289 + 0.34027 \times [1.63935 - 4.17461] = 2.16020 \text{ m/s}$$
$$u_3 = u_3^* + d_3(p'_C - p'_D) = 3.50087 + 0.23437 \times [4.17461 - 6.20805] = 3.02428 \text{ m/s}$$
$$u_4 = u_4^* + d_4(p'_D - p'_E) = 4.10926 + 0.15 \times [6.20805 - 0.0] = 5.04047 \text{ m/s}$$

We can also calculate the corrected nodal pressure at A using equation (6.65):

$$p_A = p_0 - \tfrac{1}{2}\rho u_1^2 (A_1/A_A)^2 = 10 - \tfrac{1}{2} \times 1.0 \times (1.68015 \times 0.45/0.5)^2 = 8.85671$$

First, we check whether the velocity field satisfies continuity. The mass flow rates ρuA calculated at u-nodes are

Continuity check				
Node	1	2	3	4
ρuA	0.75607	0.75607	0.75607	0.75607

The exact mass flow rate from Bernoulli's equation is 0.44721 kg/s, so the error in the computed mass flow rate is +69%. This is not a problem, because we would not expect the solution after one iteration to be accurate. Nevertheless, the fact that the mass flow rate at all four velocity nodes is exactly the same highlights an important feature of SIMPLE, which also applies in more complex multi-dimensional problems: the algorithm aims to supply a continuity-satisfying velocity field at the end of each iteration cycle. The close link with this key conservation principle has proved to be a major strength of the SIMPLE algorithm and its variants.

The computed velocity solution at the end of an iteration cycle is not yet in balance with the computed pressure field, i.e. momentum is not yet conserved. Of course this is due to the fact that the entries in the discretised momentum equations were computed on the basis of an assumed initial velocity field. Therefore, we need to perform iterations until both continuity and momentum equations are satisfied.

Under-relaxation

In the iteration process the SIMPLE algorithm requires under-relaxation. For the next iteration we use under-relaxation factors for both velocity and pressure (say 0.8 for both) and start the next solution cycle with the following velocity and pressure fields:

$$u_{new} = (1 - 0.8) \times u_{old} + 0.8 \times u_{calculated}$$

$$p_{new} = (1 - 0.8) \times p_{old} + 0.8 \times p_{calculated}$$

The velocity field for the next iteration is

u_1 m/s	u_2 m/s	u_3 m/s	u_4 m/s
1.78856	2.29959	3.21942	5.36571

As explained in section 6.4, equations (6.36) and (6.37), a_P, S_u and d values of the discretised momentum equations are also under-relaxed. Note that, for illustration purposes, these under-relaxation measures were not included in the a_P, S_u and d values of the discretised momentum equations shown earlier. In practice under-relaxation is used from the start of the calculation: hence the resulting solution with the above under-relaxation measures would be slightly different from the one shown.

Iterative convergence and residuals

If we substitute the under-relaxed velocity and pressure fields into the discretised momentum equations they will not satisfy the equations unless by chance we have computed the final solution in one iteration (e.g. due to

a fortunate choice of initial velocity and pressure fields). For example, the discretised momentum equation at node 1 in the next iteration is

$$1.20425u_1 = 1.98592$$

The difference between the left and right hand sides of the discretised momentum equation at every velocity node is called the **momentum residual**. Substituting the current velocity value of $u_1 = 1.78856$ yields:

u-momentum residual at node 1 = $1.20425 \times 1.78856 - 1.98592 = 0.16795$

If the iteration sequence is convergent this residual should decrease to show an improving balance between the computed velocity and pressure fields. Ideally, we would like to stop the iteration process when mass and momentum are exactly balanced in the discretised pressure correction and momentum equations. In practice, the finite precision of number representation in computing machinery would make this impossible and, even if it were possible to compute with very high precision, this would take far too much computing time. Our aim is to truncate the iterative sequence when we are sufficiently close to exact balance that further improvement is of limited practical value.

We calculate momentum residuals at all velocity nodes and monitor the sum of *absolute* values of the residuals as an indication of satisfactory progress of the calculation sequence. We note that residuals can be positive as well as negative. Using the sum of absolute values prevents spurious indication of convergence due to cancellation between positive and negative contributions. We accept the solution when the sum of absolute residuals is less than a predetermined small value (say 10^{-5}). It should be noted that this is a weak test to determine the point where the iterative sequence can be truncated. A decreasing sum of residuals could be due to residuals that decrease by roughly the same amount at every node or due to a small number of decreasing residuals in conjunction with others that do not decrease much at all. In a grid with a large number of nodes a few increasing residuals might even be hidden amongst a much larger number of strongly decreasing residuals. Nevertheless, summed residuals calculations are routinely used as convergence indicator in fluid flow calculations. For a further discussion of the use of residuals and iterative convergence the reader is referred to Chapter 10.

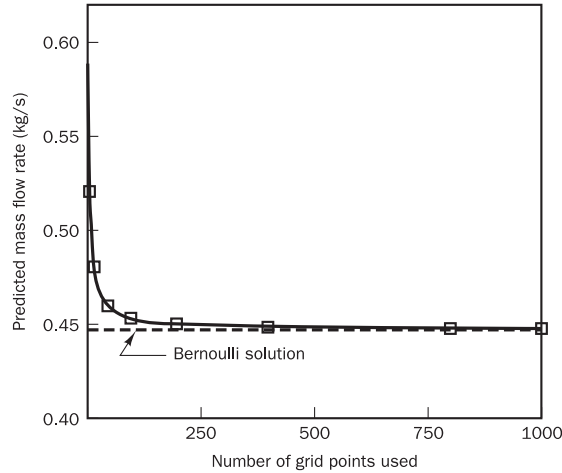
Solution

Application of under-relaxation factors of 0.8 for both u and p and allowing a maximum sum of absolute momentum residuals of 10^{-5} yields convergence in 19 iterations. The solution is given in the table below

Converged pressure and velocity field after 19 iterations

Pressure (Pa)				***Velocity (m/s)***			
Node	*Computed*	*Exact*	*Error (%)*	*Node*	*Computed*	*Exact*	*Error (%)*
A	9.22569	9.60000	−3.9	1	1.38265	0.99381	39.1
B	9.00415	9.37500	−4.0	2	1.77775	1.27775	39.1
C	8.25054	8.88889	−7.2	3	2.48885	1.78885	39.1
D	6.19423	7.50000	−17.4	4	4.14808	2.98142	39.1
E	0	0	–				

The converged mass flow rate for this five-node grid is 0.62221 kg/s, which is 39% higher than the exact value. By refining the grid we can get progressively closer solution to the exact solution. Using grids with 10, 20 and 50 nodal points yields mass flow rates of 0.5205, 0.4805 and 0.4597 kg/s, respectively. This illustrates how the error in the solution can be reduced by systematic grid refinement. If the grid is further refined to say 200, 500 and 1000 grid nodes the computed mass flow rate converges to the exact solution of 0.44721 kg/s. This is graphically illustrated in Figure 6.12.

Figure 6.12 Predicted mass flow rate with different grids

Some comments on exit boundary conditions

Finally, we briefly discuss the issue of the downstream boundary condition. In Example 6.1 the exit pressure was set to $p_D = 0$. Solution of the pressure correction equations yields the pressures at nodes other than D as **gauge pressures** (relative to p_D). Say the absolute pressure at D had a non-zero reference value $p_{Abs,D} = p_{ref}$ at this location. The **absolute pressure** field at node N can be found by adding the absolute pressure at D to the gauge pressure at N: $p_{Abs,N} = p_{Ref} + p_{Gauge,N}$. If the absolute pressure is known at some other reference location R in the computational domain ($p_{Abs,R} = p_{Ref}$) the absolute pressure at node N can be obtained by means of $p_{Abs,N} = p_{Ref} + (p_{Gauge,N} - p_{Gauge,R})$. In constant-property flows the actual value of p_{ref} is immaterial, since pressure differences appear in the source terms in the discretised momentum equations. When we solve problems involving fluids with properties that depend on the absolute pressure (e.g. compressible flows) we modify the SIMPLE algorithm by including one more iterative structures to update the fluid properties as new absolute pressures become available.

As we have seen in section 2.10 it is also possible to use an outflow boundary condition at the downstream boundary in conjunction with a given inlet velocity. The outflow boundary condition requires the gradient of the velocity to be zero at the downstream boundary. This can be implemented by equating the velocities at the two nodes that straddle the boundary, i.e. by setting

$$u_5 = u_4 \tag{6.70}$$

In Example 6.1 we have seen that a fixed pressure boundary condition is implemented by setting the pressure correction to zero, which reduces the original system of pressure correction equations by one equation. Equation (6.70) cannot provide a pressure, so if this zero velocity gradient boundary condition was used in Example 6.1 we would have only three pressure correction equations for nodes A, B and C, but four (unknown) pressure corrections p'_A, p'_B, p'_C, p'_D. Thus, it appears as if we are one equation short. To overcome this problem we note again that pressure differences are important in the discretised momentum equation and use the above device of setting an arbitrary reference pressure at the exit plane: $p_D = p_{ref}$. For the sake of expediency it is easiest to set $p_D = p_{ref} = 0$. Having fixed the pressure we may set the pressure correction equal to zero, solve the pressure correction equation as in the above examples and obtain the pressure solution as a gauge pressure.

6.11 Summary

The most popular solution algorithms for pressure and velocity calculations with the finite volume method have been discussed. They all possess the following common characteristics:

- The problems associated with the non-linearity of the momentum equations and the coupling between transport equations are tackled by adopting an iterative solution strategy.
- Velocity components are defined on staggered grids to avoid problems associated with pressure field oscillations of high spatial frequency.
- In the staggered grid arrangement velocities are stored at the cell faces of scalar control volumes. The discretised momentum equations are solved on staggered control volumes whose cell faces contain the pressure nodes.
- The SIMPLE algorithm is an iterative procedure for the calculation of pressure and velocity fields. Starting from an initial pressure field p^* its principal steps are:
  - solve discretised momentum equation to yield intermediate velocity field (u^*, v^*)
  - solve the continuity equation in the form of an equation for pressure correction p'
  - correct pressure and velocity by means of

$$p_{I,J} = p^*_{I,J} + p'_{I,J}$$
$$u_{i,J} = u^*_{i,J} + d_{i,J}(p'_{I-1,J} - p'_{I,J})$$
$$v_{I,j} = v^*_{I,j} + d_{I,j}(p'_{I,J-1} - p'_{I,J})$$

  - solve all other discretised transport equations for scalars ϕ
  - repeat until p, u, v and ϕ fields have all converged.
- Refinements to SIMPLE have produced more economical and stable iteration methods.
- The steady state PISO algorithm adds an extra correction step to SIMPLE to enhance its performance per iteration.
- It is unclear which of the procedures is the best for general-purpose computation.
- Under-relaxation is required in all methods to ensure stability of the iteration process.

Chapter seven Solution of discretised equations

7.1 Introduction

In the previous chapters we have discussed methods of discretising the governing equations of fluid flow and heat transfer. This process results in a system of linear algebraic equations which needs to be solved. The complexity and size of the set of equations depends on the dimensionality of the problem, the number of grid nodes and the discretisation practice. Although any valid procedure can be used to solve the algebraic equations, the available computer resources set a powerful constraint. There are two families of solution techniques for linear algebraic equations: **direct methods** and **indirect or iterative methods**. Simple examples of direct methods are Cramer's rule matrix inversion and Gaussian elimination. The number of operations to the solution of a system of N equations with N unknowns by means of a direct method can be determined beforehand and is of the order of N^3. The simultaneous storage of all N^2 coefficients of the set of equations in core memory is required.

Iterative methods are based on the repeated application of a relatively simple algorithm leading to eventual convergence after a – sometimes large – number of repetitions. Well-known examples are the Jacobi and Gauss–Seidel point-iterative methods. The total number of operations, typically of the order of N per iteration cycle, cannot be predicted in advance. Stronger still, it is not possible to guarantee convergence unless the system of equations satisfies fairly exacting criteria. The main advantage of iterative solution methods is that only non-zero coefficients of the equations need to be stored in core memory.

The one-dimensional conduction example in Chapter 4, section 4.3, led to a tri-diagonal system – a system with only three non-zero coefficients per equation. When QUICK differencing is applied to a convection–diffusion problem this gives rise to a penta-diagonal system that has five non-zero coefficients, which is somewhat more complex to deal with. Nevertheless, the finite volume method usually yields systems of equations, each of which has a vast majority of zero entries. Since the systems arising from realistic CFD problems can be very large – up to 100 000 to 1 million equations – we find that iterative methods are generally much more economical than direct methods.

Thomas (1949) developed a technique for rapidly solving tri-diagonal systems that is now called the Thomas algorithm or the tri-diagonal matrix algorithm (TDMA). The TDMA is actually a direct method for one-dimensional situations, but it can be applied iteratively, in a line-by-line fashion, to solve multi-dimensional problems and is widely used in CFD programs. It is computationally inexpensive and has the advantage that it

requires a minimum amount of storage. In this chapter the TDMA is explained in detail in sections 7.2 to 7.5.

The Jacobi and Gauss–Seidel methods are general-purpose point-iterative algorithms that are easily implementable, but their convergence rate can be slow when the system of equations is large. They were not initially considered suitable for general-purpose CFD procedures. However, more recently multigrid acceleration techniques have been developed that have improved the convergence rates of iterative solvers to such an extent that they are now the method of choice in commercial CFD codes. Point-iterative techniques and multigrid accelerators will be described in sections 7.6 and 7.7. This chapter closes with a brief discussion of alternative methods.

7.2 The TDMA

Consider a system of equations that has a tri-diagonal form

$$\phi_1 = C_1 \quad (7.1a)$$
$$-\beta_2\phi_1 + D_2\phi_2 - \alpha_2\phi_3 = C_2 \quad (7.1b)$$
$$-\beta_3\phi_2 + D_3\phi_3 - \alpha_3\phi_4 = C_3 \quad (7.1c)$$
$$-\beta_4\phi_3 + D_4\phi_4 - \alpha_4\phi_5 = C_4 \quad \vdots$$
$$\cdots = \cdot \quad \vdots$$
$$-\beta_n\phi_{n-1} + D_n\phi_n - \alpha_n\phi_{n+1} = C_n \quad (7.1n)$$
$$\phi_{n+1} = C_{n+1} \quad (7.1n+1)$$

In the above set of equations ϕ_1 and ϕ_{n+1} are known boundary values. The general form of any single equation is

$$-\beta_j\phi_{j-1} + D_j\phi_j - \alpha_j\phi_{j+1} = C_j \quad (7.2)$$

Equations (7.1b–n) of the above set can be rewritten as

$$\phi_2 = \frac{\alpha_2}{D_2}\phi_3 + \frac{\beta_2}{D_2}\phi_1 + \frac{C_2}{D_2} \quad (7.3a)$$

$$\phi_3 = \frac{\alpha_3}{D_3}\phi_4 + \frac{\beta_3}{D_3}\phi_2 + \frac{C_3}{D_3} \quad (7.3b)$$

$$\phi_4 = \frac{\alpha_4}{D_4}\phi_5 + \frac{\beta_4}{D_4}\phi_3 + \frac{C_4}{D_4} \quad (7.3c)$$

$$\vdots$$

$$\phi_n = \frac{\alpha_n}{D_n}\phi_{n+1} + \frac{\beta_n}{D_n}\phi_{n-1} + \frac{C_n}{D_n} \quad (7.3n–1)$$

These equations can be solved by forward elimination and back-substitution. The **forward elimination** process starts by removing ϕ_2 from equation (7.3b) by substitution from equation (7.3a) to give

$$\phi_3 = \left(\frac{\alpha_3}{D_3 - \beta_3\dfrac{\alpha_2}{D_2}}\right)\phi_4 + \left[\frac{\beta_3\left(\dfrac{\beta_2}{D_2}\phi_1 + \dfrac{C_2}{D_2}\right) + C_3}{D_3 - \beta_3\dfrac{\alpha_2}{D_2}}\right] \quad (7.4a)$$

If we adopt the notation

$$A_2 = \frac{\alpha_2}{D_2} \quad \text{and} \quad C_2' = \frac{\beta_2}{D_2}\phi_1 + \frac{C_2}{D_2} \tag{7.4b}$$

equation (7.4a) can be written as

$$\phi_3 = \left(\frac{\alpha_3}{D_3 - \beta_3 A_2}\right)\phi_4 + \left(\frac{\beta_3 C_2' + C_3}{D_3 - \beta_3 A_2}\right) \tag{7.4c}$$

If we let

$$A_3 = \frac{\alpha_3}{D_3 - \beta_3 A_2} \quad \text{and} \quad C_3' = \frac{\beta_3 C_2' + C_3}{D_3 - \beta_3 A_2}$$

equation (7.4c) can be recast as

$$\phi_3 = A_3\phi_4 + C_3' \tag{7.5}$$

Formula (7.5) can now be used to eliminate ϕ_3 from (7.3c) and the procedure can be repeated up to the last equation of the set. This constitutes the forward elimination process.

For the **back-substitution** we use the general form of recurrence relationship (7.5):

$$\phi_j = A_j\phi_{j+1} + C_j' \tag{7.6a}$$

where

$$A_j = \frac{\alpha_j}{D_j - \beta_j A_{j-1}} \tag{7.6b}$$

$$C_j' = \frac{\beta_j C_{j-1}' + C_j}{D_j - \beta_j A_{j-1}} \tag{7.6c}$$

The formulae can be made to apply at the boundary points $j = 1$ and $j = n + 1$ by setting the following values for A and C':

$$A_1 = 0 \text{ and } C_1' = \phi_1$$

$$A_{n+1} = 0 \text{ and } C_{n+1}' = \phi_{n+1}$$

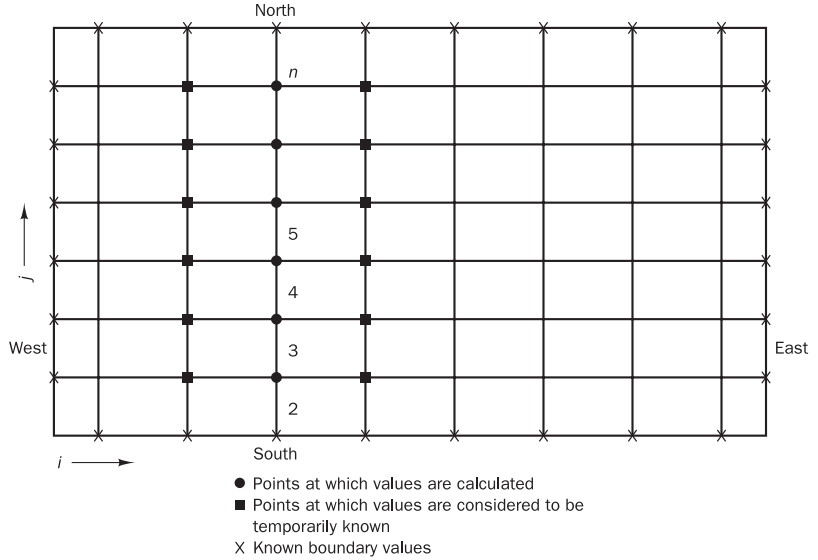
In order to solve a system of equations it is first arranged in the form of equation (7.2) and α_j, β_j, D_j and C_j are identified. The values of A_j and C_j' are subsequently calculated starting at $j = 2$ and going up to $j = n$ using (7.6b–c). Since the value of ϕ is known at boundary location $(n + 1)$ the values for ϕ_j can be obtained in reverse order (ϕ_n, ϕ_{n-1}, ϕ_{n-2}, . . . , ϕ_2) by means of the recurrence formula (7.6a). The method is simple and easy to incorporate into CFD programs. A Fortran subroutine for the TDMA is given in Anderson *et al*. (1984).

In the above derivation of the TDMA we assumed that boundary values ϕ_1 and ϕ_{n+1} were given. To implement a fixed gradient (or flux) boundary condition, e.g. at $j = 1$, the coefficient β_2 in equation (7.1b) is set to zero and the flux across the boundary is incorporated in source term C_2. The actual value of the variable at the boundary is now not directly used in the formulation. The absence of the first or the last value does not pose a problem in applying the TDMA, as will be illustrated in examples below.

7.3 Application of the TDMA to two-dimensional problems

The TDMA can be applied iteratively to solve a system of equations for two-dimensional problems. Consider the grid in Figure 7.1 and a general two-dimensional discretised transport equation of the form

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N + b \tag{7.7}$$

Figure 7.1 Line-by-line application of the TDMA

To solve the system the TDMA is applied along chosen, e.g. north–south (*n–s*), lines. The discretised equation is rearranged in the form

$$-a_S\phi_S + a_P\phi_P - a_N\phi_N = a_W\phi_W + a_E\phi_E + b \tag{7.8}$$

The right hand side of (7.8) is assumed to be temporarily known. Equation (7.8) is in the form of equation (7.2) where $\alpha_j \equiv a_N$, $\beta_j \equiv a_S$, $D_j \equiv a_P$ and $C_j \equiv a_W\phi_W + a_E\phi_E + b$. Now we can solve along the *n–s* direction of the chosen line for values $j = 2, 3, 4, \ldots, n$ as shown in Figure 7.1.

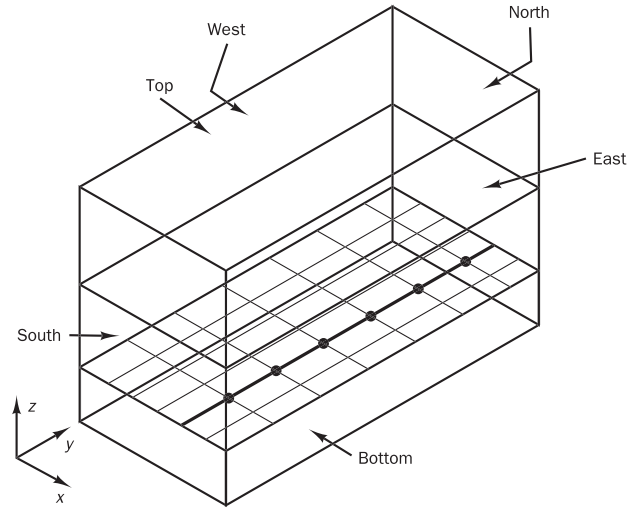
Subsequently the calculation is moved to the next north–south line. The sequence in which lines are moved is known as the sweep direction. If we sweep from west to east the values of ϕ_W to the west of a point P are known from the calculations on the previous line. Values of ϕ_E to its east, however, are unknown so the solution process must be iterative. At each iteration cycle ϕ_E is taken to have its value at the end of the previous iteration or a given initial value (e.g. zero) at the first iteration. The line-by-line calculation procedure is repeated several times until a converged solution is obtained.

7.4 Application of the TDMA to three-dimensional problems

For three-dimensional problems the TDMA is applied line by line on a selected plane and then the calculation is moved to the next plane, scanning the domain plane by plane. For example, to solve along a *n–s* line in the *x–y* plane of Figure 7.2, a discretised transport equation is written as

$$-a_S\phi_S + a_P\phi_P - a_N\phi_N = a_W\phi_W + a_E\phi_E + a_B\phi_B + a_T\phi_T + b \tag{7.9}$$

The values at W and E as well as those at B and T on the right hand side of Equation (7.9) are considered to be temporarily known. Using the TDMA, values of ϕ along a selected north–south line are computed. The calculation is moved to the next line and subsequently swept through the whole plane until all unknown values on each line have been calculated. After completion of one plane the process is moved on to the next plane.

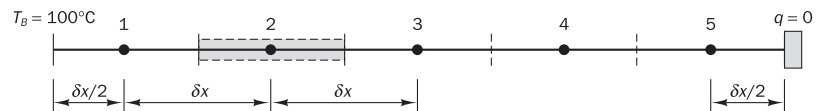
Figure 7.2 Application of the TDMA in a three-dimensional geometry

In two- and three-dimensional computations the convergence can often be accelerated by **alternating the sweep direction** so that all boundary information is fed into the calculation more effectively. To solve along an east–west line in the present three-dimensional case the discretised equation is rearranged as follows:

$$-a_W\phi_W + a_P\phi_P - a_E\phi_E = a_S\phi_S + a_N\phi_N + a_B\phi_B + a_T\phi_T + b \qquad (7.10)$$

7.5 Examples

Example 7.1 An illustration of the TDMA in one dimension

We consider the one-dimensional steady state conductive/convective heat transfer from a bar of material discussed first in Example 4.3 of section 4.3. The geometry is shown in Figure 7.3. The temperature on the left hand boundary is taken to be 100°C and the right hand side boundary is insulated so the heat flux across it is zero. Heat is lost to the surroundings by convective heat transfer. Solve the matrix equation (4.49) for this problem using the TDMA.

Figure 7.3 The grid for Example 7.1

Solution

The matrix equation found in section 4.3 was

$$\begin{bmatrix} 20 & -5 & 0 & 0 & 0 \\ -5 & 15 & -5 & 0 & 0 \\ 0 & -5 & 15 & -5 & 0 \\ 0 & 0 & -5 & 15 & -5 \\ 0 & 0 & 0 & -5 & 10 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \end{bmatrix} = \begin{bmatrix} 1100 \\ 100 \\ 100 \\ 100 \\ 100 \end{bmatrix} \qquad (4.49)$$

The general form of the equation used in the TDMA is

$$-\beta_j\phi_{j-1} + D_j\phi_j - \alpha_j\phi_{j+1} = C_j \qquad (7.2)$$

Nodes 1 and 5 are boundary nodes so we set $\beta_1 = 0$ and $\alpha_5 = 0$. The ϕ at the boundaries is not used; the boundary conditions enter into the calculation through the source terms C_j.

To show the results most clearly the values of α, β, D and C are given for each node in Table 7.1, and A_j and C'_j, calculated using the recurrence formulae (7.6b) and (7.6c), are given in Table 7.2.

Table 7.1

Node	β_j	D_j	α_j	C_j	A_j	C'_j
1	0	20	5	1100	0.25	55
2	5	15	5	100	0.3636	27.2727
3	5	15	5	100	0.3793	17.9308
4	5	15	5	100	0.3816	14.4735
5	5	10	0	100	0.00	21.3009

Table 7.2 Specimen calculation

$$A_j = \frac{\alpha_j}{D_j - \beta_j A_{j-1}} \qquad C'_j = \frac{\beta_j C'_{j-1} + C_j}{D_j - \beta_j A_{j-1}}$$

$$A_1 = \frac{5}{(20-0)} = 0.25 \qquad C'_1 = \frac{0 + 1100}{(20-0)} = 55$$

$$A_2 = \frac{5}{(15 - 5 \times 0.25)} = 0.3636 \qquad C'_2 = \frac{5 \times 55 + 100}{(15 - 5 \times 0.25)} = 27.2727$$

$$A_3 = \frac{5}{(15 - 5 \times 0.3636)} = 0.3793 \qquad C'_3 = \frac{5 \times 27.2727 + 100}{(15 - 5 \times 0.3636)} = 17.9308$$

$$A_4 = \frac{5}{(15 - 5 \times 0.3793)} = 0.3816 \qquad C'_4 = \frac{5 \times 17.9308 + 100}{(15 - 5 \times 0.3793)} = 14.4735$$

$$A_5 = 0 \qquad C'_5 = \frac{5 \times 14.4735 + 100}{(10 - 5 \times 0.3816)} = 21.3009$$

Solution with the back-substitution formula (7.6a), $\phi_j = A_j \phi_{j+1} + C'_j$, gives

$$\phi_5 = 0 + 21.30 = 21.30$$

$$\phi_4 = 0.3816 \times 21.30 + 14.4735 = 22.60$$

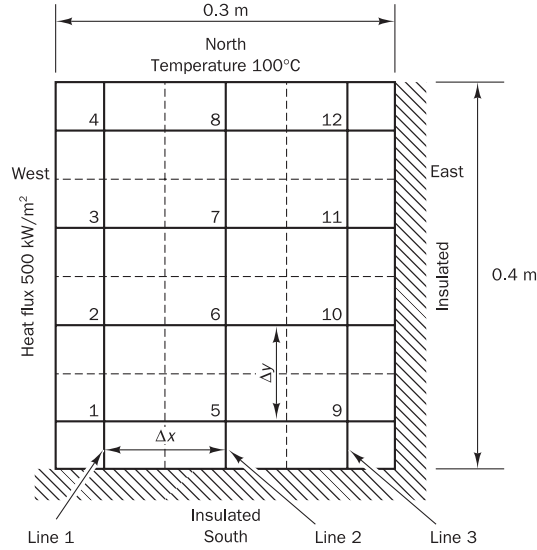
$$\phi_3 = 0.3793 \times 22.60 + 17.9308 = 26.50$$

$$\phi_2 = 0.3636 \times 26.50 + 27.2727 = 36.91$$

$$\phi_1 = 0.25 \times 36.91 + 55 = 64.23$$

Example 7.2 A two-dimensional line-by-line application of the TDMA

In Figure 7.4 a two-dimensional plate of thickness 1 cm is shown. The thermal conductivity of the plate material is k = 1000 W/m.K. The west boundary receives a steady heat flux of 500 kW/m^2 and the south and east boundaries are insulated. If the north boundary is maintained at a temperature of 100°C, use a uniform grid with $\Delta x = \Delta y = 0.1$ m to calculate the steady state temperature distribution at nodes 1, 2, 3, 4, . . . etc.

Figure 7.4 Boundary conditions for the two-dimensional heat transfer problem described in Example 7.2

Solution

The two-dimensional steady state heat transfer in the plate is governed by

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) = 0 \tag{7.11}$$

This can be written in discretised form as

$$a_P T_P = a_W T_W + a_E T_E + a_S T_S + a_N T_N \tag{7.12a}$$

where

$$a_W = \frac{k}{\Delta x}A_w \quad a_E = \frac{k}{\Delta x}A_e \quad a_S = \frac{k}{\Delta y}A_s \quad a_N = \frac{k}{\Delta y}A_n \tag{7.12b}$$

$$a_P = a_W + a_E + a_S + a_N \tag{7.12c}$$

In this case, the values of all neighbour coefficients are equal:

$$a_W = a_E = a_N = a_S = \frac{1000}{0.1} \times (0.1 \times 0.01) = 10$$

At interior points 6 and 7

$$a_P = a_W + a_E + a_S + a_N = 40$$

so the discretised equation at point 6 is

$$40T_6 = 10T_2 + 10T_{10} + 10T_5 + 10T_7$$

All nodes except 6 and 7 are adjacent to boundaries.

At a boundary node the discretised equation takes the form

$$a_P T_P = a_W T_W + a_E T_E + a_S T_S + a_N T_N + S_u$$
$$a_P = a_W + a_E + a_S + a_N - S_p$$

The boundary conditions are incorporated into the discretised equations by setting the relevant coefficient to zero and by the inclusion of source terms through S_u and S_p. Otherwise, the procedure is the same as in the one-dimensional Example 7.1. We demonstrate the approach by forming the discretised equations for boundary nodes 1 and 4.

At node 1

West is a constant flux boundary; let b_W be the contribution to the source term from the west:

$$a_W = 0$$
$$b_W = q_w \, . \, A_w = 500 \times 10^3 \times (0.1 \times 0.01) = 500$$

South is an insulated boundary; no flux enters the control volume through the south boundary:

$$a_S = 0$$
$$b_S = 0$$

Total source

$$S_u = b_W + b_S = 500$$
$$S_p = 0$$

The discretised equation at node 1 is

$$20T_1 = 10T_2 + 10T_5 + 500$$

At node 4

West is a constant flux boundary

$$a_W = 0$$
$$b_W = 500 \times 10^3 \times (0.1 \times 0.01) = 500$$

North is a constant temperature boundary

$$a_N = 0$$

$$b_N = \frac{2k}{\Delta y} A_n \times 100 = 2000$$

$$S_{P_N} = -\frac{2k}{\Delta y} A_n = -20$$

Total source

$$S_u = b_W + b_N = 500 + 2000 = 2500$$
$$S_p = -20$$

Now we have

$$a_p = a_S + a_E - S_P = 10 + 10 + 20 = 40$$
$$S_u = 2500$$

The discretised equation at node 4 is

$$40T_4 = 10T_3 + 10T_8 + 2500$$

The coefficients and the source term of the discretisation equation for all points are summarised in Table 7.3.

Table 7.3

Node	a_N	a_S	a_W	a_E	a_P	S_u
1	10	0	0	10	20	500
2	10	10	0	10	30	500
3	10	10	0	10	30	500
4	0	10	0	10	40	2500
5	10	0	10	10	30	0
6	10	10	10	10	40	0
7	10	10	10	10	40	0
8	0	10	10	10	50	2000
9	10	0	10	0	20	0
10	10	10	10	0	30	0
11	10	10	10	0	30	0
12	0	10	10	0	40	2000

Let us apply the TDMA along north–south lines, sweeping from west to east. The discretisation equation is given by

$$-a_S T_S + a_P T_P - a_N T_N = a_W T_W + a_E T_E + b \tag{7.13}$$

For convenience the line in Figure 7.4 containing points 1 to 4 is referred to as line 1, the one containing points 5 to 8 as line 2, and the one with points 9 to 12 as line 3. All west coefficients are zero at points 1, 2, 3 and 4: hence the values to the west of line 1 do not enter into the calculation. East values (points 5, 6, 7 and 8) are required for the evaluation of C. They are unknown at this stage and are assumed to be zero as an initial guess. The values of α_j, β_j, D_j and C_j can be calculated using equations (7.2) and (7.13). Now we have $\alpha_j = a_N$, $\beta_j = a_S$, $D_j = a_P$ and $C_j = a_W T_W + a_E T_E + S_u$. The values of α_j, β_j, D_j and C_j and A_j and C'_j for line 1 are summarised in Table 7.4 and the calculations for A_j and C'_j in Table 7.5.

Table 7.4

Node	β_j	D_j	α_j	C_j	A_j	C'_j
1	0	20	10	500	0.5	25
2	10	30	10	500	0.4	30
3	10	30	10	500	0.385	30.769
4	10	40	0	2500	0	77.667

Table 7.5

$A_j = \dfrac{\alpha_j}{D_j - \beta_j A_{j-1}}$	$C'_j = \dfrac{\beta_j C'_{j-1} + C_j}{D_j - \beta_j A_{j-1}}$
$A_1 = \dfrac{10}{(20-0)} = 0.5$	$C'_1 = \dfrac{0 + 500}{(20-0)} = 25$
$A_2 = \dfrac{10}{(30 - 10 \times 0.5)} = 0.4$	$C'_2 = \dfrac{10 \times 25 + 500}{(30 - 10 \times 0.5)} = 30$
$A_3 = \dfrac{10}{(30 - 10 \times 0.4)} = 0.385$	$C'_3 = \dfrac{10 \times 30 + 500}{(30 - 10 \times 0.4)} = 30.769$
$A_4 = 0$	$C'_4 = \dfrac{10 \times 30.769 + 2500}{(40 - 10 \times 0.385)} = 77.667$

Solution by back-substitution gives

$$T_4 = 0 + 77.667 \\ = 77.67$$

$$T_3 = 0.385 \times 77.667 + 30.769 \\ = 60.67$$

$$T_2 = 0.4 \times 60.67 + 30 \\ = 54.27$$

$$T_1 = 0.5 \times 54.268 + 25 \\ = 52.13$$

The TDMA calculation procedure for line 2 is similar to line 1. Here the values to the west are known from the calculations given above and the values to the east are assumed to be zero. We leave the detailed calculations as an exercise for the reader. The values of α_j, β_j, D_j and C_j for points 5, 6, 7 and 8 are summarised in Table 7.6.

Table 7.6

Node	β_j	D_j	α_j	C_j
5	0	30	10	521.3
6	10	40	10	542.6
7	10	40	10	606.5
8	10	50	0	2776.7

The TDMA solution for line 2 is $T_5 = 27.38$, $T_6 = 30.03$, $T_7 = 38.47$ and $T_8 = 63.23$. We can now proceed to the third line containing points 9, 10, 11 and 12. The values of α_j, β_j, D_j and C_j are summarised in Table 7.7.

Table 7.7

Node	β_j	D_j	α_j	C_j
9	0	20	10	273.8
10	10	30	10	300.3
11	10	30	10	384.7
12	10	40	0	2632.3

At the end of the first iteration we have the values shown in Table 7.8 for the entire field.

Table 7.8 Values at the end of first iteration

Node	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
T	52.13	54.27	60.67	77.67	27.38	30.03	38.47	63.23	32.79	38.21	51.82	78.76

The entire procedure is now repeated until a converged solution is obtained. In this case after 37 iterations we obtain the converged solution (total error less than 1.0) shown in Table 7.9.

Table 7.9 The converged solution after 37 iterations

Node	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
T	260.0	242.2	205.6	146.3	222.7	211.1	178.1	129.7	212.1	196.5	166.2	124.0

7.5.1 Closing remarks

We have discussed the solution of systems of equations with the TDMA. This algorithm is highly economical for tri-diagonal systems. It consists of a forward elimination and a back-substitution stage:

- *Forward elimination*
  - arrange system of equations in the form of (7.2):

    $$-\beta_j\phi_{j-1} + D_j\phi_j - \alpha_j\phi_{j+1} = C_j$$

  - calculate coefficients α_j, β_j, D_j and C_j
  - starting at $j = 2$ calculate A_j and C'_j using (7.6b–c):

    $$A_j = \alpha_j/(D_j - \beta_j A_{j-1})^{-1} \quad \text{and} \quad C'_j = (\beta_j C'_{j-1} + C_j)/(D_j - \beta_j A_{j-1})^{-1}$$

  - repeat for $j = 3$ to $j = n$
- *Back-substitution*
  - starting at $j = n$ obtain ϕ_n by evaluating (7.6a):

    $$\phi_j = A_j\phi_{j+1} + C'_j$$

  - repeat for $j = n - 1$ to $j = 2$ giving ϕ_{n-1} to ϕ_2 in reverse order

For two- and three-dimensional problems, the TDMA must be applied iteratively in a line-by-line fashion, but the spread of boundary information into the calculation domain can be slow. In CFD calculations the convergence rate depends on the sweep direction, with sweeping from upstream to downstream along the flow direction producing faster convergence than sweeping against the flow or parallel to the flow direction. Convergence problems can be alleviated by alternating the sweep directions, which is particularly useful in complex three-dimensional recirculating flows, where the dominant flow direction is not known in advance. When overall stability considerations require coupling between the values over the whole calculation domain the TDMA can be unsatisfactory for the solution of discretised equations.

Higher-order schemes for the discretisation process will link each discretisation equation to nodes other than the immediate neighbours. Here, the TDMA can only be applied by incorporating several neighbouring contributions in the source term. This may be undesirable in terms of stability, can impair the effectiveness of the higher-order scheme, and may hinder the implicit nature of the scheme if it is applied in an unsteady flow (see Chapter 8). In the specific case where the system of equations to be solved has the form of a penta-diagonal matrix, as may be the case in QUICK and other higher-order discretisation schemes, there is an alternative solution: a generalised version of the TDMA, known as the penta-diagonal matrix algorithm, is available. Basically a sequence of operations is carried out on the original matrix to reduce it to upper triangular form, and back-substitution is performed to obtain the solution. Details of the method can be found in Fletcher (1991). The method is, however, not nearly as economical as the TDMA.

7.6 Point-iterative methods

Point-iterative techniques are introduced by means of a simple example. Consider a set of three equations and three unknowns:

$$\begin{aligned} 2x_1 &+ x_2 &+ x_3 &= 7 \\ -x_1 &+ 3x_2 &- x_3 &= 2 \\ x_1 &- x_2 &+ 2x_3 &= 5 \end{aligned} \tag{7.14}$$

In iterative methods we rearrange the first equation to place x_1 on the left hand side, the second equation to get x_2 on the left hand side, and so on. This yields

$$\begin{aligned} x_1 &= (7 - x_2 - x_3)/2 \\ x_2 &= (2 + x_1 + x_3)/3 \\ x_3 &= (5 - x_1 + x_2)/2 \end{aligned} \tag{7.15}$$

We see that unknowns x_1, x_2 and x_3 appear on both sides of (7.15). The system of equations can be solved iteratively by substituting a set of guessed initial values for x_1, x_2 and x_3 on the right hand side. This allows us to calculate new values of the unknowns on the left hand side of (7.15). The next move is to substitute the new values back into the right hand side and evaluate the unknowns on the left hand side again, which are, if the procedure converges, closer to the true solution of the system of equations. This process is repeated until there is no more change in the solution.

One condition for the iteration process to be convergent is that the matrix must be diagonally dominant (see discussion on boundedness in

section 5.4.2). When general systems of equations are solved it is sometimes necessary to rearrange the equations, but the finite volume method yields diagonally dominant systems as part of the discretisation process, so this aspect does not require special attention.

The Jacobi and Gauss–Seidel methods apply slightly different substitutions on the right hand side. Below we describe the main features both methods.

7.6.1 Jacobi iteration method

In the Jacobi method, the values $x_1^{(k)}$, $x_2^{(k)}$ etc. on the left hand side at iteration (k) – indicated here by the bracketed superscript – are evaluated by substituting in the right hand side the last known values $x_1^{(k-1)}$, $x_2^{(k-1)}$ etc., which were obtained at iteration ($k - 1$). In the above example, let us use $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$ as the initial guess. Substitution of these values in the right hand side of (7.15) gives

$$x_1^{(1)} = 3.500 \quad x_2^{(1)} = 0.667 \quad x_3^{(1)} = 2.500$$

For the second iteration we substitute these values in the right hand side of (7.15). If we repeat the process we obtain the results given in Table 7.10.

Table 7.10 Solution of system of equations (7.14) with Jacobi method

Iteration number	*0*	*1*	*2*	*3*	*4*	*5*	*. . .*	*17*
x_1	0	3.5000	1.9167	1.6250	1.2292	1.1563	. . .	1.0000
x_2	0	0.6667	2.6667	1.6667	2.1667	1.9167	. . .	2.0000
x_3	0	2.5000	1.0833	2.8750	2.5208	2.9688	. . .	3.0000

After 17 iterations we obtain $x_1 = 1.0000$, $x_2 = 2.0000$, $x_3 = 3.0000$ and detect no further change in the solution with increase of the iteration count. Substitution of these values into the original system (7.14) shows that this result is accurate to all 4 decimal places given in the answer.

To generalise the procedure we consider a system of n equations and n unknowns in matrix form, $\mathbf{A} \,.\, \mathbf{x} = \mathbf{b}$, or in a form where the coefficients of matrix $\mathbf{A}$ can be seen explicitly:

$$\sum_{j=1}^{n} a_{ij} x_j = b_i \tag{7.16}$$

In all iterative methods the system is rearranged to place the contribution due to x_i on the left hand side of the ith equation and the other terms on the right hand side:

$$a_{ii} x_i = b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} x_j \quad (i = 1, 2, \ldots, n) \tag{7.17}$$

We divide both sides by coefficient a_{ii} and indicate that, in the Jacobi method, we evaluate the left hand side at iteration (k) using values on the right hand side of x_j at the end of the previous iteration ($k - 1$):

$$\boxed{x_i^{(k)} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \left(\frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}}} \quad (i = 1, 2, \ldots, n) \tag{7.18}$$

Equation (7.18) is the **iteration equation for the Jacobi method** in the form used for actual calculations. In matrix form this equation can be written as follows:

$$\mathbf{x}^{(k)} = \mathbf{T} \, . \, \mathbf{x}^{(k-1)} + \mathbf{c} \tag{7.19a}$$

where $\mathbf{T}$ = iteration matrix
and $\mathbf{c}$ = constant vector

The coefficients T_{ij} of the **iteration matrix** are as follows:

$$T_{ij} = \begin{cases} -\dfrac{a_{ij}}{a_{ii}} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \tag{7.19b}$$

and the elements of the constant vector are

$$c_i = \frac{b_i}{a_{ii}} \tag{7.19c}$$

7.6.2 Gauss–Seidel iteration method

We begin our discussion of the Gauss–Seidel method by reconsidering equation (7.15). In the Jacobi method the right hand side is evaluated using the results of the previous iteration level or from the initial guess. If all the right hand sides could be evaluated simultaneously there would be no further discussion, but in most computing machines the calculations are performed sequentially. Hence, at the first iteration we start the sequence of calculations by using the initial guesses $x_2^{(0)} = 0$ and $x_3^{(0)} = 0$ to obtain

$$x_1^{(1)} = (7 - x_2^{(0)} - x_3^{(0)})/2 = (7 - 0 - 0)/2 = 3.5$$

Next we evaluate the second equation, $x_2 = (2 + x_1 + x_3)/3$. We notice that it contains x_1 and x_3 on the right hand side. The Jacobi method uses $x_1^{(0)} = 0$ and $x_3^{(0)} = 0$ from the initial guesses, but we note that in a sequential calculation we have just obtained an updated value of x_1, namely $x_1^{(1)} = 3.5$. The Gauss–Seidel method proceeds by making direct use of this recently available value and calculates

$$x_2^{(1)} = (2 + x_1^{(1)} + x_3^{(0)})/3 = (2 + 3.5 + 0)/3 = 1.8333$$

To evaluate the third equation, $x_3 = (5 - x_1 + x_2)/2$, the Gauss–Seidel method continues to use the most up-to-date values on the right hand side that are available, i.e. $x_1^{(1)} = 3.5$ and $x_2^{(1)} = 1.8333$:

$$x_3^{(1)} = (5 - x_1^{(1)} + x_2^{(1)})/2 = (5 - 3.5 + 1.8333)/2 = 1.6667$$

The second and subsequent iterations follow the same pattern. The results are shown in Table 7.11.

Table 7.11 Solution of system of equations (7.14) with Gauss–Seidel method

Iteration number	*0*	*1*	*2*	*3*	*4*	*5*	*. . .*	*13*
x_1	0	3.5000	1.7500	1.3333	1.1181	1.0475	. . .	1.0000
x_2	0	1.8333	1.8056	1.9537	1.9761	1.9922	. . .	2.0000
x_3	0	1.6667	2.5278	2.8102	2.9290	2.9724	. . .	3.0000

The final result is obtained after 13 iterations. Ralston and Rabinowitz (1978) note that the Gauss–Seidel method is preferable to the Jacobi method, because it converges faster.

We can easily generalise the above example and state the **iteration equation for the Gauss–Seidel method**:

$$\boxed{x_i^{(k)} = \sum_{j=1}^{i-1}\left(\frac{-a_{ij}}{a_{ii}}\right)x_j^{(k)} + \sum_{j=i+1}^{n}\left(\frac{-a_{ij}}{a_{ii}}\right)x_j^{(k-1)} + \frac{b_i}{a_{ii}}} \quad (i = 1, 2, \ldots, n) \tag{7.20}$$

In matrix form we have

$$\mathbf{x}^{(k)} = \mathbf{T}_1\mathbf{x}^{(k)} + \mathbf{T}_2\mathbf{x}^{(k-1)} + \mathbf{c} \tag{7.21a}$$

The coefficients of matrices $\mathbf{T}_1$ and $\mathbf{T}_2$ are as follows:

$$T_{1ij} = \begin{cases} -\dfrac{a_{ij}}{a_{ii}} & \text{if } i > j \\ 0 & \text{if } i \le j \end{cases} \tag{7.21b}$$

$$T_{2ij} = \begin{cases} 0 & \text{if } i \ge j \\ -\dfrac{a_{ij}}{a_{ii}} & \text{if } i < j \end{cases} \tag{7.21c}$$

and the elements of the constant vector are as before:

$$c_i = \frac{b_i}{a_{ii}} \tag{7.21d}$$

7.6.3 Relaxation methods

The convergence rate of the Jacobi and Gauss–Seidel methods depends on the properties of the iteration matrix. It has been found that these can be improved by the introduction of a so-called relaxation parameter α. Consider the iteration equation (7.18) for the Jacobi method. It is easy to see that it can also be written as

$$x_i^{(k)} = x_i^{(k-1)} + \sum_{j=1}^{n}\left(\frac{-a_{ij}}{a_{ii}}\right)x_j^{(k-1)} + \frac{b_i}{a_{ii}} \quad (i = 1, 2, \ldots, n) \tag{7.22}$$

We try to modify the convergence rate of the iteration sequence by multiplying the second and third terms on the right hand side by the **relaxation parameter** α:

$$x_i^{(k)} = x_i^{(k-1)} + \alpha\left[\sum_{j=1}^{n}\left(\frac{-a_{ij}}{a_{ii}}\right)x_j^{(k-1)} + \frac{b_i}{a_{ii}}\right] \quad (i = 1, 2, \ldots, n) \qquad (7.23)$$

If we use $\alpha = 1$ in (7.23) we get back to the original Jacobi method (7.18), but different values of parameter α will yield different iterative sequences. When we choose $0 < \alpha < 1$ the procedure is an **under-relaxation** method, whereas $\alpha > 1$ is called **over-relaxation**.

Before proceeding to apply (7.23) we verify that introduction of the relaxation parameter α changes the iteration path without changing the final solution. First we compare the expression in the square brackets of (7.23) with matrix equation (7.16). If the iteration sequence converges, the vector $x_j^{(k\to\infty)}$ will contain the correct solution of the system, so

$$\sum_{j=1}^{n} a_{ij}x_j^{(k\to\infty)} = b_i \qquad (i = 1, 2, \ldots, n)$$

Dividing both sides by coefficient a_{ii} and some rearrangement yields

$$\frac{b_i}{a_{ii}} + \sum_{j=1}^{n}\frac{-a_{ij}}{a_{ii}}x_j^{(k\to\infty)} = 0 \quad (i = 1, 2, \ldots, n) \qquad (7.24)$$

After k iterations the intermediate solution vector $x_j^{(k)}$ is not equal to the correct solution, so

$$\sum_{j=1}^{n} a_{ij}x_j^{(k)} \neq b_i \qquad (i = 1, 2, \ldots, n) \qquad (7.25)$$

We define the **residual** $r_i^{(k)}$ of the ith equation after k iterations as the difference between the left and right hand sides of (7.25):

$$r_i^{(k)} = b_i - \sum_{j=1}^{n} a_{ij}x_j^{(k)} \qquad (i = 1, 2, \ldots, n) \qquad (7.26)$$

If the iteration process is convergent the intermediate solution vector $x_j^{(k)}$ should get progressively closer to final solution vector $x_j^{(k\to\infty)}$ as the iteration count k increases, and hence the residuals $r_i^{(k)}$ for all n equations should also tend to zero as $k \to \infty$. Finally, we note that the expression in the square brackets of (7.23) is just equal to the residual $r_i^{(k-1)}$ after $k - 1$ iterations divided by coefficient a_{ii}:

$$x_i^{(k)} = x_i^{(k-1)} + \alpha\left[\frac{r_i^{(k-1)}}{a_{ii}}\right] \quad (i = 1, 2, \ldots, n) \qquad (7.27)$$

This confirms that the introduction of relaxation parameter α does not affect the converged solution, because all residuals $r_i^{(k-1)}$ in the square brackets of (7.27) will be zero when $k \to \infty$.

Next we note that, in terms of the iteration matrix form (7.19a–c) of the equation, the introduction of the relaxation parameter in (7.23) implies the

following changes to the coefficients T_{ij} of the iteration matrix and constant vector c_i:

$$T_{ij} = \begin{cases} -\alpha \dfrac{a_{ij}}{a_{ii}} & \text{if } i \neq j \\ (1-\alpha) & \text{if } i = j \end{cases} \tag{7.28a}$$

$$c_i = \alpha \frac{b_i}{a_{ii}} \tag{7.28b}$$

Thus, we have demonstrated that the relaxation parameter alters the iteration path through changes in the iteration matrix, without altering the final solution. This suggests that relaxation may be advantageous if we select an optimum value of α that minimises the number of iterations required to reach the converged solution.

To see if this works in practice we perform the Jacobi iteration scheme with relaxation (7.23) for the example system (7.14) using the same initial guess as before: $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0$ with $\alpha = 0.75$, 1.0 and 1.25. We find that the process converges to the correct solution $x_1 = 1$, $x_2 = 2$, $x_3 = 3$ after 25, 17 and 84 iterations, respectively. It appears that $\alpha = 1$ is the optimum value for the Jacobi method and that there is not much to be gained by changes of α (at least not for this sample problem).

In spite of this slightly disappointing result we try out the relaxation concept on the Gauss–Seidel method. In this case the iteration equation after k iterations can be rewritten as

$$x_i^{(k)} = x_i^{(k-1)} + \sum_{j=1}^{i-1} \left(\frac{-a_{ij}}{a_{ii}} \right) x_j^{(k)} + \sum_{j=i}^{n} \left(\frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}}$$

$$(i = 1, 2, \ldots, n)$$

If we introduce the relaxation parameter α as before, this yields

$$\boxed{x_i^{(k)} = x_i^{(k-1)} + \alpha \left[\sum_{j=1}^{i-1} \left(\frac{-a_{ij}}{a_{ii}} \right) x_j^{(k)} + \sum_{j=i}^{n} \left(\frac{-a_{ij}}{a_{ii}} \right) x_j^{(k-1)} + \frac{b_i}{a_{ii}} \right]}$$

$$(i = 1, 2, \ldots, n) \tag{7.29}$$

This is the iteration sequence for the **Gauss–Seidel method with relaxation**. We leave it as an exercise for the reader to verify that iteration of the sequence (7.29) using coefficients and the right hand side of example system (7.14) with $\alpha = 0.75$, 1.0 and 1.25 yields convergence after 21, 13 and 27 iterations, respectively. It seems, once again, that no improvement is possible, but a slightly more careful search reveals that the iteration sequence converges to 4 decimal places within 10 iterations for slightly over-relaxed values of α in the range 1.06 to 1.08.

Unfortunately, the optimum value of the relaxation parameter is problem and mesh dependent, and it is difficult to give precise guidance. Nevertheless, through experience with a particular range of similar problems it is, at least in principle, possible to select a value of α which gives a better convergence rate than the basic Gauss–Seidel method. The well-known **successive over-relaxation** (SOR) technique is based on this principle.

7.7 Multigrid techniques

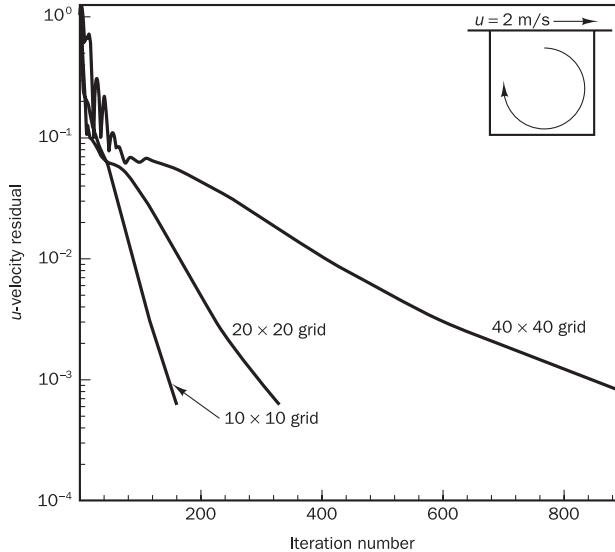
We have established in earlier chapters that the discretisation error reduces with the mesh spacing. In other words, the finer the mesh, the better the accuracy of a CFD simulation. Iterative techniques are preferred over direct methods because their storage overheads are lower, which makes them more attractive for the solution of large systems of equations arising from highly refined meshes. Moreover, we have seen in Chapter 6 that the SIMPLE algorithm for the coupling of continuity and momentum equations is itself iterative. Hence, there is no need to obtain very accurate intermediate solutions, as long as the iteration process eventually converges to the true solution. Unfortunately, it transpires that the **convergence rate of iterative methods**, such as the Jacobi and Gauss–Seidel, rapidly **reduces as the mesh is refined**.

To examine the relationship between the convergence rate of an iterative method and the number of grid cells in a problem we consider a simple two-dimensional cavity-driven flow. The inset of Figure 7.5 shows that the computational domain is a square cavity with a size of 1 cm × 1 cm. The lid of the cavity is moving with a velocity of 2 m/s in the positive x-direction. The fluid in the cavity is air and the flow is assumed to be laminar. We use a line-by-line iterative solver to compute the solution on three different grids with 10 × 10, 20 × 20 and 40 × 40 cells.

To obtain a measure of the closeness to the true solution of an intermediate solution in an iteration sequence we use the residual defined in (7.26) for the ith equation. The **average residual** $\bar{r}$ over all n equations in the system (i.e. an average over all the control volumes in the computational domain of a flow problem) is a useful indicator of iterative convergence for a given problem:

$$\bar{r} = \frac{1}{n}\sum_{i=1}^{n}|r_i| \tag{7.30}$$

If the iteration process is convergent the average residual $\bar{r}$ should tend to zero, since all contributing residuals $r_i \to 0$ as $k \to \infty$. The average residual

Figure 7.5 Residual reduction pattern with a line-by-line iterative solver using different grid resolutions

for a given solution parameter, e.g. the u-velocity component, is usually normalised to make it easier to interpret its value from case to case and to compare it with residuals relating to other solution parameters (e.g. v- or w-velocity or pressure, which may each have very different magnitudes). The most common **normalisation** is to consider the ratio of the average residual after k iterations and its value at the first iteration:

$$R_{norm}^{(k)} = \frac{\bar{r}^{(k)}}{\bar{r}^{(1)}} \tag{7.31}$$

In Figure 7.5 we have plotted the normalised residual of the u-momentum equation against the iteration number. The solution is aborted when the normalised residuals for all solution variables (velocity and pressure in this case) fall below 10^{-3}. We note that the 10×10 mesh solution converges in 161 iterations, whereas the 20×20 and 40×40 mesh solutions take 331 and 891 iterations to converge, respectively. Within the CFD code it is possible to improve the convergence rate by adjusting solution parameters, including relaxation parameters, but for the sake of consistency all solution parameters were kept constant. The pattern of residual reduction is evident from the diagram. After a rapid initial reduction of the residuals their rate of decrease settles to a more modest final value. It is also clear that the **final convergence rate is lowest for the finest mesh**. If we tried an even finer mesh, it would take even longer to converge.

Multigrid concept

To simplify the explanation of the multigrid method we use matrix notation and first revisit the definition of the residual. Consider the following system of equations arising from the finite volume discretisation of a conservation equation on a flow domain:

$$\mathbf{A} \,.\, \mathbf{x} = \mathbf{b} \tag{7.32}$$

The vector $\mathbf{x}$ is the true solution of system (7.32).

If we solve this system with an iterative method we obtain an intermediate solution $\mathbf{y}$ after some unspecified number of iterations. This intermediate solution does not satisfy (7.32) exactly and, as before, we define the residual vector $\mathbf{r}$ as follows:

$$\mathbf{A} \,.\, \mathbf{y} = \mathbf{b} - \mathbf{r} \tag{7.33}$$

We can also define an error vector $\mathbf{e}$ as the difference between the true solution and the intermediate solution:

$$\mathbf{e} = \mathbf{x} - \mathbf{y} \tag{7.34}$$

Subtracting (7.33) from (7.32) gives the following relationship between the error vector and the residual vector:

$$\mathbf{A} \,.\, \mathbf{e} = \mathbf{r} \tag{7.35}$$

The residual vector can be easily calculated at any stage of the iteration process by substituting the intermediate solution into (7.33). We can imagine using an iterative process to solve system (7.35) and obtain the error vector. For this it might be useful to write the system in the iteration matrix form:

$$\mathbf{e}^{(k)} = \mathbf{T} \,.\, \mathbf{e}^{(k-1)} + \mathbf{c} \tag{7.36a}$$

Since the coefficient matrix **A** is the same for systems (7.32) and (7.35), the coefficients T_{ij} of the iteration matrix are equal to those of the chosen iteration method, i.e. the Jacobi method or Gauss–Seidel method without or with relaxation. The elements of the constant vector are, however, different:

$$c_i = \frac{r_i}{a_{ii}} \tag{7.36b}$$

In practice, if we tried to solve system (7.35) using the same iteration method as we used for the original system (7.32) we would not find that this made any difference in terms of convergence rate. However, system (7.35) is important, because it shows how the error propagates from one iteration to the next. Moreover, its equivalent (7.36) highlights the crucial role played by the iteration matrix. As we saw earlier when we introduced the relaxation technique, the properties of the iteration matrix determine the rate of error propagation and, hence, the rate of convergence.

These properties have been extensively studied along with the mathematical behaviour of the error propagation as a function of the iterative technique, mesh size, discretisation scheme etc. It has been established that the solution error has components with a range of wavelengths that are multiples of the mesh size. Iteration methods cause rapid reduction of error components with short wavelengths up to a few multiples of the mesh size. However, long-wavelength components of the error tend to decay very slowly as the iteration count increases.

This error behaviour explains the observed trends in Figure 7.5. For the coarse mesh, the longest possible wavelengths of error components (i.e. those of the order of the domain size) are just within the short-wavelength range of the mesh and, hence, all error components reduce rapidly. On the finer meshes, however, the longest error wavelengths are progressively further outside the short-wavelength range for which decay is rapid.

Multigrid methods are designed to exploit these inherent differences of the error behaviour and use *iterations on meshes of different size*. The short-wavelength errors are effectively reduced on the finest meshes, whereas the long-wavelength errors decrease rapidly on the coarsest meshes. Moreover, the computational cost of iterations is larger on finer meshes than on coarse meshes, so the extra cost due to iterations on the coarse meshes is offset by the benefit of much improved convergence rate.

7.7.1 An outline of a multigrid procedure

We now give a short description of the principles of a two-stage multigrid procedure:

Step 1: *Fine grid iterations*. Perform iterations on the finest grid with mesh spacing h to generate an intermediate solution $\mathbf{y}^h$ to system $\mathbf{A}^h \cdot \mathbf{x} = \mathbf{b}$ with true solution vector $\mathbf{x}$. The number of iterations is chosen sufficiently large that the short-wavelength oscillatory component of the error is effectively reduced, but no attempt is made to eliminate the long-wavelength error component. The residual vector $\mathbf{r}^h$ for the solution on this mesh satisfies $\mathbf{r}^h = \mathbf{b} - \mathbf{A}^h \cdot \mathbf{y}^h$ (see equation (7.33)) and the error vector $\mathbf{e}^h$ is given by $\mathbf{e}^h = \mathbf{x} - \mathbf{y}^h$ (see equation (7.34)). We have also established that the error and residual are related as follows: $\mathbf{A}^h \cdot \mathbf{e}^h = \mathbf{r}^h$ (see equation (7.35)).

Step 2: *Restriction*. The solution is transferred from the fine mesh with spacing h onto a coarse mesh with spacing ch, where $c > 1$. Due to the larger mesh spacing of the coarse mesh the long-wavelength error (on the fine mesh) appears as a short-wavelength error on the new mesh and will reduce rapidly. The process of transferring can be simplified if we use a coarse mesh with twice the mesh spacing of the fine mesh. Instead of solving for the solution vector $\mathbf{y}^{ch}$ we work with the error equation $\mathbf{A}^{ch} \cdot \mathbf{e}^{ch} = \mathbf{r}^{ch}$ on the coarse mesh starting with an initial guess of $\mathbf{e}^{ch} = \mathbf{0}$. To perform the solution process we need values of the residual vector and the matrix of coefficients. Given the values of $\mathbf{r}^{h}$ on the fine mesh we must use a suitable averaging procedure to find the residual vector $\mathbf{r}^{ch}$ on the coarse mesh. The coefficients of matrix $\mathbf{A}^{ch}$ can be recomputed from scratch on the coarser mesh or evaluated from the fine mesh coefficient matrix $\mathbf{A}^{h}$ using some form of averaging or interpolation technique. The cost per iteration on the coarser mesh is small, so we can afford to perform an adequate number of iterations to get a converged solution of the error vector $\mathbf{e}^{ch}$.

Step 3: *Prolongation*. After obtaining the converged solution of error vector $\mathbf{e}^{ch}$ for the coarse mesh we need to transfer it back to the fine mesh, but note that we have fewer data than points in the fine mesh. We use a convenient interpolation operator (e.g. linear interpolation) to generate values for the prolonged error vector $\mathbf{e}'^{h}$ at intermediate points in the fine mesh.

Step 4: *Correction and final iterations*. Once we have calculated the prolonged error vector $\mathbf{e}'^{h}$ we may correct the intermediate fine grid solution: $\mathbf{y}^{improved} = \mathbf{y}^{h} + \mathbf{e}'^{h}$. Because the long-wavelength error has been eliminated, this improved solution is closer to the true solution vector $\mathbf{x}$. However, several approximations were made, so we perform a few more iterations with the improved solution to iron out any errors that may have been introduced during restriction and prolongation.

The above description is for the two-stage procedure (one fine mesh, one coarse mesh). In practice, however, the restriction is carried out into a number of increasingly coarse levels. Then prolongation procedures are also performed at each stage back to the starting mesh.

7.7.2 An illustrative example

Example 7.5

Consider solving a one-dimensional conduction equation for an insulated metal rod which has internal heat generation. The governing equation is

$$k\frac{d^2T}{dx^2} + g = 0$$

The dimensions and other data are as follows: length of the rod is 1 m, cross-sectional area of the rod is 0.01 m^2, thermal conductivity k = 5 W/m.K, generation g = 20 kW/m^3, the ends are at 100°C and 500°C. We are interested in obtaining a solution, say, using a grid of 20 cells giving a spacing of Δx = 0.05 m, which we name Grid 1.

Solution

Figure 7.6 shows Grid 1 along with the boundary conditions marked at each end. It is not necessary here to describe how discretisation equations

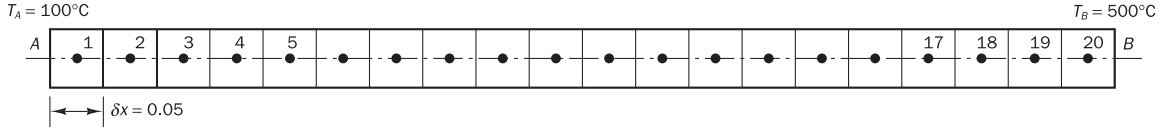



Figure 7.6 The 20 node grids used to solve the problem – Grid 1

are obtained for this problem; the procedure is similar to Example 4.2. Table 7.12 gives a summary of coefficients of the discretisation equations at nodes 1, 2, 3, . . . , 20.

Table 7.12 Coefficients of the discretisation equation at each point

Node	a_W	a_E	S_u	S_p	a_P
1 (first node)	0	$\frac{kA}{\delta x}$	$qA\delta x + \frac{2kA}{\delta x}T_A$	$-\frac{2kA}{\delta x}$	$a_W + a_E - S_p$
2, 3, . . . , 19 (internal nodes)	$\frac{kA}{\delta x}$	$\frac{kA}{\delta x}$	$qA\delta x$	0.0	$a_W + a_E - S_p$
20 (last node)	$\frac{kA}{\delta x}$	0	$qA\delta x + \frac{2kA}{\delta x}T_B$	$-\frac{2kA}{\delta x}$	$a_W + a_E - S_p$

We use the expressions in Table 7.12 to compile the numerical values of coefficients in Table 7.13 and to construct the matrix equation **A . x = b**, where solution vector **x** contains the temperatures at the nodes of Grid 1.

Table 7.13 Numerical values of the coefficients of the discretisation equation

Node	a_W	a_E	S_u	S_p	a_P
1	0	1.0	210	–2.0	3.0
2, 3, . . . , 19	1.0	1.0	10	0.0	2.0
20	1.0	0	1010	–2.0	3.0

The matrix equation is

$$\begin{bmatrix} 3.0 & -1.0 & 0 & . & . & . & 0 \\ -1.0 & 2.0 & -1.0 & . & . & . & 0 \\ 0 & -1.0 & 2.0 & -1.0 & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & -1.0 & 2.0 & -1.0 \\ . & . & . & . & . & -1.0 & 3.0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ x_{19} \\ x_{20} \end{bmatrix} = \begin{bmatrix} 210 \\ 10 \\ 10 \\ \vdots \\ \vdots \\ 10 \\ 1010 \end{bmatrix} \quad (7.37)$$

The matrix of coefficients is tri-diagonal, so we can use the TDMA to obtain a solution in a single pass. The result is given in Table 7.14 to enable later verification of the multigrid solution.

Table 7.14 The TDMA solution

Grid 1 – Temperature at nodes																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
160	270	370	460	540	610	670	720	760	790	810	820	820	810	790	760	720	670	610	540

Step 1: Fine grid iterations

We use the Gauss–Seidel iteration (7.20) to solve these equations. We simply initialise the temperature to 150°C everywhere as an initial guess to start the iterative process (a field closer to the final solution will not highlight the benefit of the method). The solution vector $\mathbf{y}^h$ after five Gauss–Seidel iterations is shown below:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ . \\ . \\ . \\ y_{19} \\ y_{20} \end{bmatrix} = \begin{bmatrix} 116.755 \\ 141.994 \\ 160.427 \\ . \\ . \\ . \\ 394.392 \\ 468.130 \end{bmatrix} \tag{7.38}$$

The residual vector $\mathbf{r}^h = \mathbf{b} - \mathbf{A}^h \,.\, \mathbf{y}^h$ at this stage is

$$\mathbf{r}^h = \begin{bmatrix} r_1^h \\ r_2^h \\ r_3^h \\ . \\ . \\ . \\ r_{19}^h \\ r_{20}^h \end{bmatrix} = \begin{bmatrix} 210 \\ 10 \\ 10 \\ . \\ . \\ . \\ 10 \\ 1010 \end{bmatrix} - \begin{bmatrix} 3.0 & -1.0 & 0 & . & . & . & 0 \\ -1.0 & 2.0 & -1.0 & . & . & . & 0 \\ 0 & -1.0 & 2.0 & -1.0 & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & -1.0 & 2.0 & -1.0 \\ . & . & . & . & . & -1.0 & 3.0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ . \\ . \\ . \\ y_{19} \\ y_{20} \end{bmatrix} = \begin{bmatrix} 1.728 \\ 3.193 \\ 4.658 \\ . \\ . \\ . \\ 7.461 \\ 0.000 \end{bmatrix}$$

The total r.m.s. residual value is 14.951. If the iteration process is continued the residual vector will reduce slowly until the convergence criterion is achieved. Figure 7.9 at the end of this section shows the pattern of convergence for the Gauss–Seidel iteration. Using a sum of r.m.s. residuals less than 10^{-6} as the convergence criterion, the final solution is achieved after 664 iterations. The converged solution is of course indistinguishable from the TDMA solution in Table 7.14.

Step 2: Restriction

To apply the multigrid method we have to construct a coarse grid first. The simplest method is to construct a grid which has half the number of cells. Figure 7.7 shows the fine mesh and the proposed coarse meshes drawn one beneath the other. The first coarse mesh uses 10 cells with a spacing of 0.1 m and is named Grid 2. The next coarse mesh – Grid 3 – consists of 5 cells with a spacing of 0.2 m.

If the fine grid mesh spacing is h, a mesh using half the number of cells would have a mesh spacing of $2h$. In the multigrid literature the mesh spacing is indicated by means of superscripts. In this notation the residual vector we have on the fine mesh is $\mathbf{r}^h$.

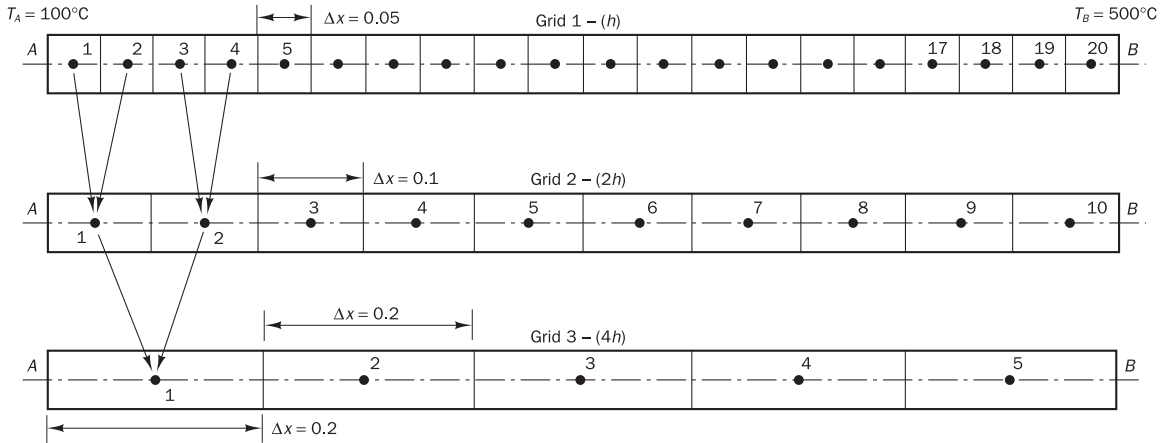



Figure 7.7 Grids used to solve the problem

Now we need to interpolate the residual vector from a fine grid to a coarse grid. Since the nodes of Grid 2 are exactly mid-way between those of Grid 1 we can interpolate by simple averaging of $\mathbf{r}^h$ to obtain the residual vector $\mathbf{r}^{2h}$ for the coarse grid. The values are summarised in Table 7.15. Note that only 3 decimal places of the actual numbers are shown in the table. As mentioned earlier, this transfer process is known as 'restriction'.

Table 7.15 Fine mesh and coarse mesh residuals showing restriction process of transfer from fine Grid 1 to coarse Grid 2

Fine mesh (Grid 1) residuals – (r^h)																			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1.728	3.193	4.658	5.976	7.075	.	.	.	.	.	.	.	.	.	.	.	33.962	22.385	7.461	0.000

Coarse mesh (Grid 2) residuals – after restriction (r^{2h})									
1	2	3	4	5	6	7	8	9	10
2.460	5.317	7.506	.	.	.	.	.	28.173	3.731

In matrix form the Grid 2 residual vector after 'restriction' is

$$\mathbf{r}^{2h} = \begin{bmatrix} r_1^{2h} \\ r_2^{2h} \\ r_3^{2h} \\ . \\ . \\ . \\ r_9^{2h} \\ r_{10}^{2h} \end{bmatrix} = \begin{bmatrix} 2.460 \\ 5.317 \\ 7.506 \\ . \\ . \\ . \\ 28.173 \\ 3.730 \end{bmatrix}$$

Note that we have only 10 values now. The error in the coarse mesh satisfies the equation $\mathbf{A}^{2h} \,.\, \mathbf{e}^{2h} = \mathbf{r}^{2h}$. We have calculated the vector $\mathbf{r}^{2h}$, but we also need the matrix $\mathbf{A}^{2h}$ to solve this equation to obtain $\mathbf{e}^{2h}$. In the multigrid literature there are numerous techniques which use elegant interpolation operators to evaluate $\mathbf{A}^{2h}$. For this example problem we do not interpolate the coefficient matrix, but calculate the coefficients of the coarse grid matrix exactly using the expressions in Table 7.12. Thus, we obtain the following matrix equation for the error vector $\mathbf{e}^{2h}$:

$$\begin{bmatrix} 1.5 & -0.5 & 0 & . & . & . & 0 \\ -0.5 & 1.0 & -0.5 & 0 & . & . & 0 \\ 0 & -0.5 & 1.0 & -0.5 & . & . & 0 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & -0.5 & 1.0 & -0.5 \\ . & . & . & . & . & -0.5 & 1.5 \end{bmatrix} \begin{bmatrix} e_1^{2h} \\ e_2^{2h} \\ e_3^{2h} \\ . \\ . \\ . \\ e_9^{2h} \\ e_{10}^{2h} \end{bmatrix} = \begin{bmatrix} 2.460 \\ 5.317 \\ 7.506 \\ . \\ . \\ . \\ 28.173 \\ 3.730 \end{bmatrix} \quad (7.39)$$

We now solve system (7.39) with an initial guess of $\mathbf{e}^{2h} = (0, 0, 0, \ldots, 0)$ using the Gauss–Seidel iteration procedure. Since the iterations are now on a coarser mesh, the rate of residual reduction is faster and the cost per iteration is much lower. After 10 iterations on this coarse mesh we obtain the error vector $\mathbf{e}^{2h}$ on the first coarse mesh (Grid 2) as

$$\begin{bmatrix} e_1^{2h} \\ e_2^{2h} \\ e_3^{2h} \\ . \\ . \\ . \\ e_9^{2h} \\ e_{10}^{2h} \end{bmatrix} = \begin{bmatrix} 19.156 \\ 58.310 \\ 96.049 \\ . \\ . \\ . \\ 158.591 \\ 55.351 \end{bmatrix} \quad (7.40)$$

Since we have only performed 10 iterations this solution is partially converged and there will be a residual: $\hat{\mathbf{r}}^{2h} = \mathbf{r}^{2h}_{\text{at start}} - \mathbf{A}^{2h} \,.\, \mathbf{e}^{2h}$. Its values are given in Table 7.16 along with interpolated Grid 3 residuals $\mathbf{r}^{4h}$ after restriction.

Table 7.16 Residuals on Grid 2 and restricted residuals on Grid 3

Coarse mesh (Grid 2) residuals $\hat{r}^{2h}$									
1	2	3	4	5	6	7	8	9	10
2.881	4.609	5.929	.	.	.	.	.	0.9192	0.000

Coarse mesh (Grid 3) residuals r^{4h} – after restriction				
1	2	3	4	5
3.745	6.277	6.204	3.615	0.459

Now the residuals $\hat{\mathbf{r}}^{2h}$ have been transferred to an even coarser grid with five nodes (see Figure 7.7) to yield residual $\mathbf{r}^{4h}$. Then we solve for the error $\mathbf{e}^{4h}$ on Grid 3 using the system of equations $\mathbf{A}^{4h} \,.\, \mathbf{e}^{4h} = \mathbf{r}^{4h}$, where the coefficients of $\mathbf{A}^{4h}$ are again calculated from scratch using the expressions in Table 7.12. As the cost per iteration is very low we can afford to do more iterations on Grid 3 to achieve very effective error reduction. After 10 iterations we get the solution in Table 7.17 for error vector $\mathbf{e}^{4h}$.

Table 7.17

Grid 3 – solution (error vector e^{4h} on Grid 3)				
1	2	3	4	5
23.408	55.831	63.731	47.205	16.348

This coarsening procedure could be continued, but in this illustrative example we stop the process of restriction at the five-node grid.

Step 3: Prolongation

The next step is to go backwards transferring error vectors from each coarse grid to the next fine grid level. This is called the prolongation process. Linear interpolation or any other interpolation scheme could be used to construct fine grid values from the coarse grid values. Using linear interpolation some sample values are

$$\begin{aligned} e'^{2h}_1 &= (0.75e^{4h}_1) \\ e'^{2h}_2 &= (0.75e^{4h}_1 + 0.25e^{4h}_2) \\ e'^{2h}_3 &= (0.25e^{4h}_1 + 0.75e^{4h}_2) \end{aligned} \tag{7.41}$$

and so on.

It should be noted that we use a prime to indicate the prolonged error vector $\mathbf{e}'^{2h}$ on Grid 2 to distinguish it from the error vector $\mathbf{e}^{2h}$. Furthermore, for the nodes closest to the boundary we have used the fact that the value of the problem variable is known so the error on the boundary is zero. The calculations of (7.41) yield the following values for the prolonged error vector $\mathbf{e}'^{2h}$:

$$\begin{bmatrix} e'^{2h}_1 \\ e'^{2h}_2 \\ e'^{2h}_3 \\ \cdot \\ \cdot \\ \cdot \\ e'^{2h}_9 \\ e'^{2h}_{10} \end{bmatrix} = \begin{bmatrix} 17.556 \\ 31.514 \\ 47.726 \\ \cdot \\ \cdot \\ \cdot \\ 24.062 \\ 12.261 \end{bmatrix}$$

The prolonged error vector $\mathbf{e}'^{2h}$ is now used to correct the original error vector $\mathbf{e}^{2h}$ (7.40) on Grid 2:

$$\mathbf{e}^{2h}_{corrected} = \mathbf{e}^{2h} + \mathbf{e}'^{2h} \tag{7.42}$$

This yields

$$\begin{bmatrix} e^{2h}_1 \\ e^{2h}_2 \\ e^{2h}_3 \\ \cdot \\ \cdot \\ \cdot \\ e^{2h}_9 \\ e^{2h}_{10} \end{bmatrix} = \begin{bmatrix} 19.156 \\ 58.310 \\ 96.049 \\ \cdot \\ \cdot \\ \cdot \\ 158.591 \\ 55.351 \end{bmatrix} + \begin{bmatrix} 17.556 \\ 31.514 \\ 47.726 \\ \cdot \\ \cdot \\ \cdot \\ 24.062 \\ 12.261 \end{bmatrix} = \begin{bmatrix} 36.713 \\ 89.825 \\ 143.775 \\ \cdot \\ \cdot \\ \cdot \\ 182.654 \\ 67.612 \end{bmatrix}$$

At this stage it is usual to do some smoothing iterations before transferring this error vector to the grid above this level. First we perform two Gauss–Seidel smoothing iterations and obtain the following corrected and smoothed error vector on Grid 2:

$$\begin{bmatrix} e^{2h}_1 \\ e^{2h}_2 \\ e^{2h}_3 \\ \cdot \\ \cdot \\ \cdot \\ e^{2h}_9 \\ e^{2h}_{10} \end{bmatrix} = \begin{bmatrix} 32.639 \\ 95.749 \\ 152.494 \\ \cdot \\ \cdot \\ \cdot \\ 188.283 \\ 65.248 \end{bmatrix}$$

This result is next used to compute the prolonged error vector $\mathbf{e}^h$ on Grid 1 by linear interpolation using the process outlined in (7.41), but replacing superscripts $2h$ by h and $4h$ by $2h$:

$$\begin{bmatrix} e_1^h \\ e_2^h \\ e_3^h \\ . \\ . \\ . \\ e_{19}^h \\ e_{20}^h \end{bmatrix} = \begin{bmatrix} 24.479 \\ 48.416 \\ 79.971 \\ . \\ . \\ . \\ 96.007 \\ 48.936 \end{bmatrix}$$

Step 4: Correction and final iterations
Finally, the prolonged error vector $\mathbf{e}^h$ is used to compute the corrected intermediate solution $\mathbf{y}$ on Grid 1:

$$\mathbf{y}_{corrected} = \mathbf{y} + \mathbf{e}^h \tag{7.43a}$$

Thus,

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ . \\ . \\ . \\ y_{19} \\ y_{20} \end{bmatrix} = \begin{bmatrix} 116.755 \\ 141.994 \\ 160.427 \\ . \\ . \\ . \\ 394.392 \\ 468.130 \end{bmatrix} + \begin{bmatrix} 24.479 \\ 48.416 \\ 79.971 \\ . \\ . \\ . \\ 96.007 \\ 48.936 \end{bmatrix} = \begin{bmatrix} 141.235 \\ 190.411 \\ 240.399 \\ . \\ . \\ . \\ 490.399 \\ 517.067 \end{bmatrix} \tag{7.43b}$$

Comparison of the corrected solution (7.43b) with intermediate solution (7.38) and the TDMA solution in Table 7.14 shows that the multigrid procedure has considerably reduced the error. Substitution of the corrected solution into $\mathbf{r} = \mathbf{b} - \mathbf{A} \,.\, \mathbf{y}$ gives an r.m.s. residual of 8.786, which is lower than the previous r.m.s. residual on Grid 1, which was 14.951. Since interpolation errors are involved in the restriction and prolongation processes we cannot expect to achieve the true solution in one multigrid cycle. In order to improve the solution further we do more iterations on the fine mesh (say two) and repeat the 'fine grid–coarse grid' procedure until convergence is achieved. We proceed by using the three-grid procedure and go backwards and forwards as many times as is needed to reduce the r.m.s. residual to 10^{-6}. This multigrid cycle is called a three-grid **V-cycle**. The process steps are illustrated in Figure 7.8 along with annotations of the number of iterations at each level inside the circles. The diagram also reveals the origin of the term 'V-cycle'.

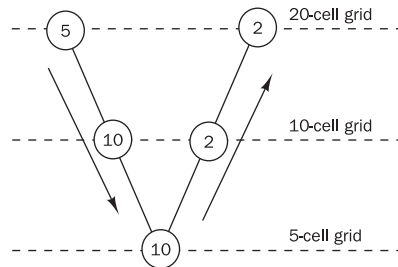
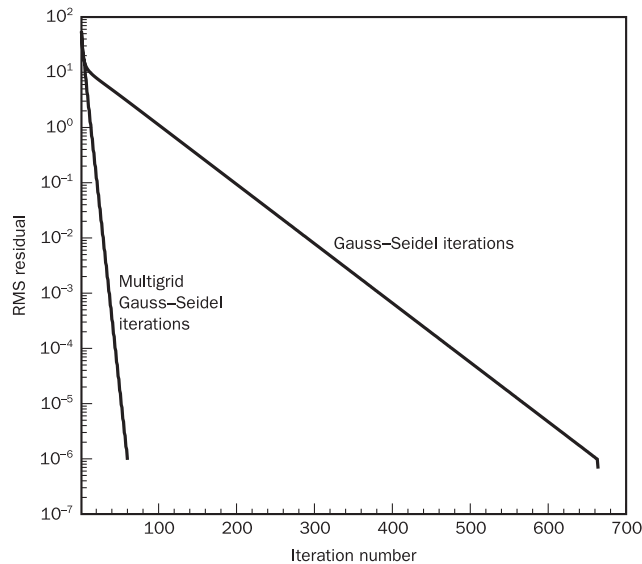
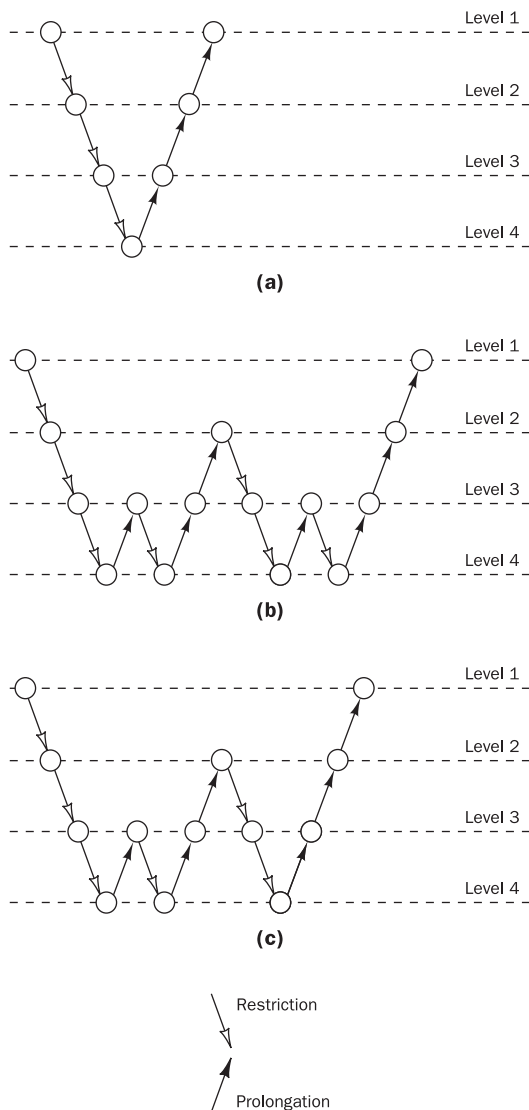
Figure 7.8 Schematic of the V-cycle multigrid procedure used in the example

The pattern of convergence achieved by repeating the V-cycle of 2 fine grid iterations, 10 and 10 coarse grid iterations is shown in Figure 7.9. The multigrid procedure is fast and effective, since it converges in 60 fine grid iterations, which compares favourably with the ordinary Gauss–Seidel method, which takes 664 iterations to achieve the same residual value. Even after allowing for extra computational effort due to the coarse grid iterations, the order-of-magnitude improvement of the convergence rate by the multigrid procedure is clearly beneficial. When multigrid acceleration techniques are applied to 2D and 3D problems the convergence gains obtained are very attractive, which explains their popularity among CFD users.

Figure 7.9 Residual reduction pattern with ordinary Gauss–Seidel iterations and multigrid Gauss–Seidel iterations

7.7.3 Multigrid cycles

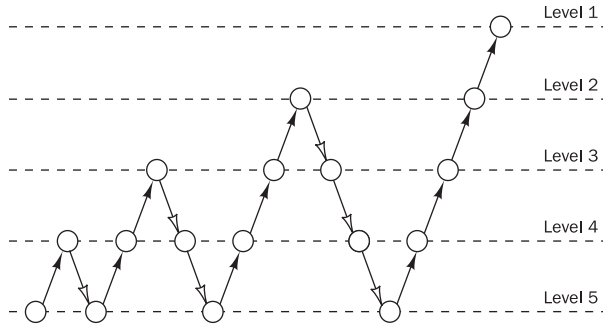
Multigrid techniques can be used in conjunction with any iterative technique. In our simple example we have illustrated the main concepts of the multigrid methods. In practical CFD calculations the multigrid transfer process is more sophisticated and different cycles of coarsening and refinement are used with special schedules of restriction and prolongation at different refinement levels. Common choices of multigrid cycles are the so-called V-, W- and F-cycles, which are illustrated in Figure 7.10.

The simple V-cycle shown in Figure 7.10a consists of two legs. The calculation starts at the finest level. Iterations at any level are called relaxation. After a few relaxation sweeps on the finest level the residuals are restricted to the next coarse level and after relaxation on that level the residuals are passed on to the next coarse level, and so on until the coarsest level is reached. After final relaxation on the coarsest level the prolongation steps are performed on the upward leg of the V-cycle until the finest level is reached.

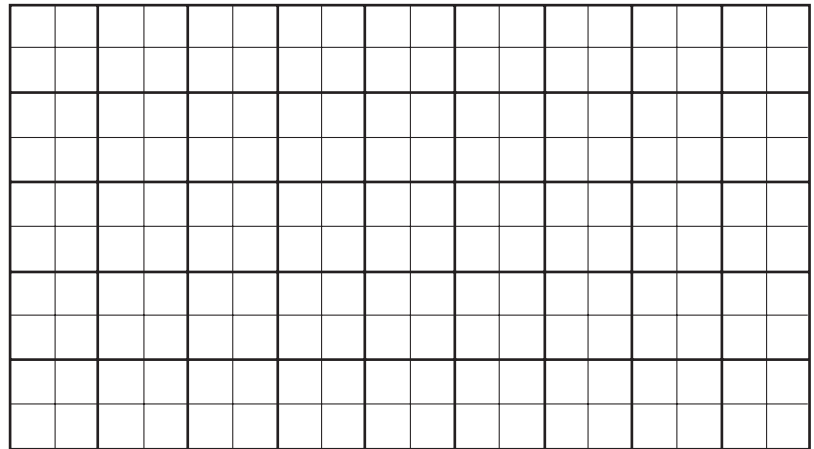
In the W-cycle additional restriction and prolongation sweeps are used at coarser levels to obtain better reduction of long-wavelength errors. A typical

Figure 7.10 Illustration of different multigrid cycle strategies: (a) V-cycle, (b) W-cycle and (c) F-cycle

pattern is illustrated in Figure 7.10b. The flexible cycle or 'F-cycle' is very similar to the W-cycle, but has a different pattern of coarse-level sweeps as illustrated in Figure 7.10c.

In the technique known as the full multigrid (FMG) method the calculations do not start at the finest grid, but instead at the coarsest level. Solutions are transferred to successively finer grid levels and on the finest level the prolonged solution is used as the initial guess for start of the iterative process. This solution process could be accelerated further using any of the cycling procedures. For example, V-cycles could be used at each successive refinement level, as illustrated in Figure 7.11.

Figure 7.11 Cycling strategy used in full multigrid method

7.7.4 Grid generation for the multigrid method

As illustrated in the above example, grid generation is required to create the coarse grids. The most straightforward method is to combine control volumes or regenerate the mesh using half the number of nodes of the mesh above that level. For 2D structured grids, such as the Cartesian grid shown in Figure 7.12, coarse grids can be readily generated by deleting alternate grid lines. Thus one coarse grid control is constructed from every four fine grid control volumes. This can easily be extended to 3D meshes using eight fine grid control volumes per coarse grid control volume.

Figure 7.12 A 2D Cartesian mesh – the coarse grid is constructed by deleting alternate grid lines or combining groups of four control volumes

In the above example we computed the coarse grid system matrix and other required quantities using actual geometrical properties of the coarse mesh (Table 7.12). This type of multigrid procedure is called a **geometric multigrid** procedure. In the other variation to this method, the coefficients are not recomputed from the grid geometry to save calculation effort, but approximated as linear combinations of coefficients of the fine grid equations. Such multigrid methods are called **algebraic multigrid** and are widely used in commercial CFD solvers. The technique known as the **additive correction multigrid** (ACM) strategy of Hutchinson and Raithby (1986) is also a popular multigrid method used in many CFD procedures.

7.8 Summary

Multigrid acceleration of the Gauss–Seidel point-iterative method is currently the solution algorithm of choice for commercial CFD codes. The rate of convergence of this procedure can be optimised by specific choices for: (i) interpolation of the residual vector and coefficient matrix from fine to coarse meshes during restriction, (ii) interpolation of the error vector from coarse to fine meshes during prolongation, (iii) cycles of coarsening and refinement with special schedules of restriction and prolongation at different refinement levels. For further details of more advanced multigrid procedures the reader should consult the appropriate literature (see e.g. Wesseling, 1992; Briggs, 1987). There are also several excellent learning resources available on the Internet for multigrid methods: see for example the multigrid network MGNET at http://www.mgnet.org/.

Several other solution algorithms are available for CFD problems with discretised equations that contain a large number of contributions from surrounding nodes. The Strongly Implicit Procedure (SIP) due to Stone (1968), in particular with the improvements suggested by Schneider and Zedan (1981), is more suitable in this case. Details are not presented here in the interest of brevity and the interested reader is referred to Anderson *et al.* (1984). Another solution procedure which is being used in CFD calculations is the conjugate gradient method (CGM) of Hestenes and Stiefel (1952). This method is based on matrix factorisation techniques. Improvements by Reid (1971), Concus *et al.* (1976) and Kershaw (1978) ensure accelerated convergence in the CFD calculations. The CGM requires greater storage than other iterative methods described earlier. Further details of the method can also be found in Press *et al.* (1992).

Chapter eight The finite volume method for unsteady flows

8.1 Introduction

Having finished the task of developing the finite volume method for steady flows we are now in a position to consider the more complex category of time-dependent problems. The conservation law for the transport of a scalar in an unsteady flow has the general form

$$\frac{\partial}{\partial t}(\rho\phi) + \mathrm{div}(\rho\mathbf{u}\phi) = \mathrm{div}(\Gamma\ \mathrm{grad}\ \phi) + S_\phi \tag{8.1}$$

The first term of the equation represents the rate of change term and is zero for steady flows. To predict transient problems we must retain this term in the discretisation process. The finite volume integration of equation (8.1) over a control volume (CV) must be augmented with a further integration over a finite time step Δt. By replacing the volume integrals of the convective and diffusive terms with surface integrals as before (see section 2.5) and changing the order of integration in the rate of change term we obtain

$$\int_{CV}\left(\int_t^{t+\Delta t}\frac{\partial}{\partial t}(\rho\phi)\,\mathrm{d}t\right)\mathrm{d}V + \int_t^{t+\Delta t}\left(\int_A \mathbf{n}\,.\,(\rho\mathbf{u}\phi)\mathrm{d}A\right)\mathrm{d}t$$

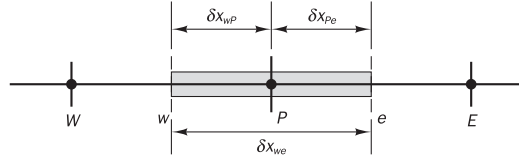
$$= \int_t^{t+\Delta t}\left(\int_A \mathbf{n}\,.\,(\Gamma\ \mathrm{grad}\ \phi)\mathrm{d}A\right)\mathrm{d}t + \int_t^{t+\Delta t}\int_{CV} S_\phi\,\mathrm{d}V\mathrm{d}t \tag{8.2}$$

So far we have made no approximations but to make progress we need techniques for evaluating the integrals. The control volume integration is essentially the same as in steady flows and the measures explained in Chapters 4 and 5 are again adopted to ensure successful treatment of convection, diffusion and source terms. Here we focus our attention on methods necessary for the time integration. The process is illustrated below using the one-dimensional unsteady diffusion (heat transfer) equation and is later extended to multi-dimensional unsteady diffusion and convection–diffusion problems.

8.2 One-dimensional unsteady heat conduction

Unsteady one-dimensional heat conduction is governed by the equation

$$\rho c\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + S \tag{8.3}$$

Figure 8.1

In addition to the usual variables we have c, the specific heat of the material (J/kg.K).

Consider the one-dimensional control volume in Figure 8.1. Integration of equation (8.3) over the control volume and over a time interval from t to $t + \Delta t$ gives

$$\int_t^{t+\Delta t}\int_{CV} \rho c \frac{\partial T}{\partial t} \mathrm{d}V\,\mathrm{d}t = \int_t^{t+\Delta t}\int_{CV} \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right)\mathrm{d}V\,\mathrm{d}t + \int_t^{t+\Delta t}\int_{CV} S\,\mathrm{d}V\,\mathrm{d}t \quad (8.4)$$

This may be written as

$$\int_w^e \left[\int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t}\mathrm{d}t\right]\mathrm{d}V = \int_t^{t+\Delta t}\left[\left(kA\frac{\partial T}{\partial x}\right)_e - \left(kA\frac{\partial T}{\partial x}\right)_w\right]\mathrm{d}t + \int_t^{t+\Delta t} \bar{S}\Delta V\,\mathrm{d}t \quad (8.5)$$

In equation (8.5), A is the face area of the control volume, ΔV is its volume, which is equal to $A\Delta x$, where $\Delta x = \delta x_{we}$ is the width of the control volume, and $\bar{S}$ is the average source strength. If the temperature at a node is assumed to prevail over the whole control volume, the left hand side can be written as

$$\int_{CV}\left[\int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t}\mathrm{d}t\right]\mathrm{d}V = \rho c(T_P - T_P^o)\Delta V \quad (8.6)$$

In equation (8.6) superscript 'o' to refers to temperatures at time t; temperatures at time level $t + \Delta t$ are not superscripted. The same result as (8.6) would be obtained by substituting $(T_P - T_P^o)/\Delta t$ for $\partial T/\partial t$, so this term has been discretised using a first-order (backward) differencing scheme. Higher-order schemes, which may be used to discretise this term, will be discussed briefly later in this chapter. If we apply central differencing to the diffusion terms on the right hand side equation (8.5) may be written as

$$\rho c(T_P - T_P^o)\Delta V = \int_t^{t+\Delta t}\left[\left(k_e A\frac{T_E - T_P}{\delta x_{PE}}\right) - \left(k_w A\frac{T_P - T_W}{\delta x_{WP}}\right)\right]\mathrm{d}t + \int_t^{t+\Delta t} \bar{S}\Delta V\,\mathrm{d}t \quad (8.7)$$

To evaluate the right hand side of this equation we need to make an assumption about the variation of T_P, T_E and T_W with time. We could use temperatures at time t or at time $t + \Delta t$ to calculate the time integral or, alternatively, a combination of temperatures at time t and $t + \Delta t$. We may generalise the approach by means of a weighting parameter θ between 0 and 1 and write the integral I_T of temperature T_P with respect to time as

$$I_T = \int_t^{t+\Delta t} T_P \, dt = [\theta T_P + (1 - \theta) T_P^o] \Delta t \tag{8.8}$$

Hence

θ	0	1/2	1
I_T	$T_P^o \Delta t$	$\frac{1}{2}(T_P + T_P^o)\Delta t$	$T_P \Delta t$

We have highlighted the following values of integral I_T: if $\theta = 0$ the temperature at (old) time level t is used; if $\theta = 1$ the temperature at new time level $t + \Delta t$ is used; and finally if $\theta = 1/2$, the temperatures at t and $t + \Delta t$ are equally weighted.

Using formula (8.8) for T_W and T_E in equation (8.7), and dividing by $A\Delta t$ throughout, we have

$$\rho c \left(\frac{T_P - T_P^o}{\Delta t} \right) \Delta x = \theta \left[\frac{k_e (T_E - T_P)}{\delta x_{PE}} - \frac{k_w (T_P - T_W)}{\delta x_{WP}} \right] + (1 - \theta) \left[\frac{k_e (T_E^o - T_P^o)}{\delta x_{PE}} - \frac{k_w (T_P^o - T_W^o)}{\delta x_{WP}} \right] + \bar{S} \Delta x \tag{8.9}$$

which may be rearranged to give

$$\left[\rho c \frac{\Delta x}{\Delta t} + \theta \left(\frac{k_e}{\delta x_{PE}} + \frac{k_w}{\delta x_{WP}} \right) \right] T_P = \frac{k_e}{\delta x_{PE}} [\theta T_E + (1 - \theta) T_E^o] + \frac{k_w}{\delta x_{WP}} [\theta T_W + (1 - \theta) T_W^o] + \left[\rho c \frac{\Delta x}{\Delta t} - (1 - \theta) \frac{k_e}{\delta x_{PE}} - (1 - \theta) \frac{k_w}{\delta x_{WP}} \right] T_P^o + \bar{S} \Delta x \tag{8.10}$$

Now we identify the coefficients of T_W and T_E as a_W and a_E and write equation (8.10) in the familiar standard form:

$$a_P T_P = a_W [\theta T_W + (1 - \theta) T_W^o] + a_E [\theta T_E + (1 - \theta) T_E^o] + [a_P^o - (1 - \theta) a_W - (1 - \theta) a_E] T_P^o + b \tag{8.11}$$

where

$$a_P = \theta (a_W + a_E) + a_P^o$$

and

$$a_P^o = \rho c \frac{\Delta x}{\Delta t}$$

with

a_W	a_E	b
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$	$\bar{S}\Delta x$

The exact form of the final discretised equation depends on the value of θ. When θ is zero, we only use temperatures T_P^o, T_W^o and T_E^o at the old time level t on the right hand side of equation (8.11) to evaluate T_P at the new time and the resulting scheme is called **explicit**. When $0 < \theta \leq 1$ temperatures at the new time level are used on both sides of the equation and the resulting schemes are called **implicit**. The extreme case of $\theta = 1$ is termed **fully implicit** and the case corresponding to $\theta = 1/2$ is called the **Crank–Nicolson** scheme (Crank and Nicolson, 1947).

8.2.1 Explicit scheme

In the explicit scheme the source term is linearised as $b = S_u + S_p T_P^o$. Now the substitution of $\theta = 0$ into (8.11) gives the **explicit discretisation** of the unsteady conductive heat transfer equation

$$a_P T_P = a_W T_W^o + a_E T_E^o + [a_P^o - (a_W + a_E - S_p)]T_P^o + S_u \tag{8.12}$$

where

$$a_P = a_P^o$$

and

$$a_P^o = \rho c \frac{\Delta x}{\Delta t}$$

a_W	a_E
$\frac{k_w}{\delta x_{WP}}$	$\frac{k_e}{\delta x_{PE}}$

The right hand side of equation (8.12) only contains values at the old time step so the left hand side can be calculated by forward marching in time. The scheme is based on backward differencing and its Taylor series truncation error accuracy is first-order with respect to time. As explained in Chapter 5, all coefficients need to be positive in the discretised equation. The coefficient of T_P^o may be viewed as the neighbour coefficient connecting the values at the old time level to those at the new time level. For this coefficient to be positive we must have $a_P^o - a_W - a_E > 0$. For constant k and uniform grid spacing, $\delta x_{PE} = \delta x_{WP} = \Delta x$, this condition may be written as

$$\rho c \frac{\Delta x}{\Delta t} > \frac{2k}{\Delta x} \tag{8.13a}$$

or

$$\boxed{\Delta t < \rho c \frac{(\Delta x)^2}{2k}} \tag{8.13b}$$

This inequality sets a stringent maximum limit to the time step size and represents a serious limitation for the explicit scheme. It becomes very expensive to improve spatial accuracy because the maximum possible time step needs to be reduced as the square of Δx. Consequently, this method is not recommended for general transient problems. Explicit schemes with greater formal accuracy than the above one have been designed. Examples are the Richardson and DuFort–Frankel methods, which use temperatures at more than two time levels. These methods also have fewer stability restrictions than the ordinary explicit method. Details of such schemes can be found in Abbot and Basco (1990), Anderson *et al* (1984) and Fletcher (1991). Nevertheless, provided that the time step size is chosen with care, the explicit scheme described above is efficient for simple conduction calculations. This will be illustrated through an example in section 8.3.

8.2.2 Crank–Nicolson scheme

The Crank–Nicolson method results from setting $\theta = 1/2$ in equation (8.11). The source term is linearised as $b = S_u + \frac{1}{2}S_P T_P + \frac{1}{2}S_P T_P^o$. Now the discretised unsteady heat conduction equation is

$$\boxed{\begin{aligned} a_P T_P = a_E\left[\frac{T_E + T_E^o}{2}\right] + a_W\left[\frac{T_W + T_W^o}{2}\right] \\ + \left[a_P^o - \frac{a_E}{2} - \frac{a_W}{2}\right] T_P^o + S_u + \frac{1}{2} S_p T_P^o \end{aligned}} \tag{8.14}$$

where

$$\boxed{a_P = \frac{1}{2}(a_W + a_E) + a_P^o - \frac{1}{2}S_p}$$

and

$$\boxed{a_P^o = \rho c \frac{\Delta x}{\Delta t}}$$

a_W	a_E
$\dfrac{k_w}{\delta x_{WP}}$	$\dfrac{k_e}{\delta x_{PE}}$

Since more than one unknown value of T at the new time level is present in equation (8.14) the method is implicit, and simultaneous equations for all node points need to be solved at each time step. Although schemes with $\frac{1}{2} \le \theta \le 1$, including the Crank–Nicolson scheme, are unconditionally stable for all values of the time step (Fletcher, 1991), it is more important to ensure that all coefficients are positive for physically realistic and bounded results. This is the case if the coefficient of T_P^o satisfies the following condition:

$$a_P^o > \left[\frac{a_E + a_W}{2} \right]$$

which leads to

$$\boxed{\Delta t < \rho c \frac{\Delta x^2}{k}} \tag{8.15}$$

This time step limitation is only slightly less restrictive than (8.13) associated with the explicit method. The Crank–Nicolson method is based on central differencing and hence it is second-order accurate in time. With sufficiently small time steps it is possible to achieve considerably greater accuracy than with the explicit method. The overall accuracy of a computation depends also on the spatial differencing practice so the Crank–Nicolson scheme is normally used in conjunction with spatial central differencing.

8.2.3 The fully implicit scheme

When the value of θ is set equal to 1 we obtain the fully implicit scheme. Now the source term is linearised as $b = S_u + S_P T_P$. The discretised equation is

$$\boxed{a_P T_P = a_W T_W + a_E T_E + a_P^o T_P^o + S_u} \tag{8.16}$$

where

$$\boxed{a_P = a_P^o + a_W + a_E - S_p}$$

and

$$\boxed{a_P^o = \rho c \frac{\Delta x}{\Delta t}}$$

with

a_W	a_E
$\dfrac{k_w}{\delta x_{WP}}$	$\dfrac{k_e}{\delta x_{PE}}$

Both sides of the equation contain temperatures at the new time step, and a system of algebraic equations must be solved at each time level (see Example 8.2). The time marching procedure starts with a given initial field

of temperatures T^O. The system of equations (8.16) is solved after selecting time step Δt. Next the solution T is assigned to T^O and the procedure is repeated to progress the solution by a further time step.

It can be seen that all coefficients are positive, which makes the implicit scheme unconditionally stable for any size of time step. Since the accuracy of the scheme is only first-order in time, small time steps are needed to ensure the accuracy of results. The implicit method is recommended for general-purpose transient calculations because of its robustness and unconditional stability.

8.3 Illustrative examples

We now demonstrate the properties of the explicit and implicit discretisation schemes by means of a comparison of numerical results for a one-dimensional unsteady conduction example with analytical solutions to assess the accuracy of the methods.

Example 8.1

A thin plate is initially at a uniform temperature of 200°C. At a certain time $t = 0$ the temperature of the east side of the plate is suddenly reduced to 0°C. The other surface is insulated. Use the explicit finite volume method in conjunction with a suitable time step size to calculate the transient temperature distribution of the slab and compare it with the analytical solution at time (i) $t = 40$ s, (ii) $t = 80$ s and (iii) $t = 120$ s. Recalculate the numerical solution using a time step size equal to the limit given by (8.13) for $t = 40$ s and compare the results with the analytical solution. The data are: plate thickness $L = 2$ cm, thermal conductivity $k = 10$ W/m.K and $\rho c = 10 \times 10^6$ J/m^3.K.

Solution

The one-dimensional transient heat conduction equation is

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k \frac{\partial T}{\partial x}\right) \tag{8.17}$$

and the initial conditions are

$$T = 200 \text{ at } t = 0$$

and the boundary conditions are

$$\frac{\partial T}{\partial x} = 0 \text{ at } x = 0,\ t > 0$$

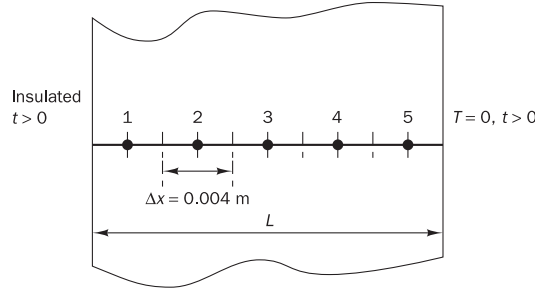
$$T = 0 \text{ at } x = L,\ t > 0$$

The analytical solution is given in Özişik (1985) as

$$\frac{T(x, t)}{200} = \frac{4}{\pi}\sum_{n=1}^{\infty}\frac{(-1)^{n+1}}{2n-1}\exp(-\alpha\lambda_n^2 t)\cos(\lambda_n x) \tag{8.18}$$

$$\text{where } \lambda_n = \frac{(2n-1)\pi}{2L} \text{ and } \alpha = k/\rho c$$

The numerical solution with the explicit method is generated by dividing the domain width L into five equal control volumes with $\Delta x = 0.004$ m. The resulting one-dimensional grid is shown in Figure 8.2.

Figure 8.2 Geometry for Example 8.1

The discretised form of governing equation (8.17) for an internal control volume using the explicit method is given by (8.12). Control volumes 1 and 5 adjoin boundaries, so the links are cut in the direction of the boundary and the boundary fluxes are included in the source terms. At the control volume 1, the west boundary is insulated: hence the flux across that boundary is zero. We modify equation (8.9) where the physics can be most easily discerned. The discretised equation at node 1 becomes

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[\frac{k}{\Delta x}(T_E^o - T_P^o) \right] - 0 \tag{8.19}$$

For time $t > 0$, the temperature of the east boundary of control volume 5 is constant (say T_B). The discretised equation at node 5 becomes

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[\frac{k}{\Delta x/2}(T_B - T_P^o) \right] - \left[\frac{k}{\Delta x}(T_P^o - T_W^o) \right] \tag{8.20}$$

All discretised equations can now be written in standard form:

$$a_P T_P = a_W T_W^o + a_E T_E^o + [a_P^o - (a_W + a_E)]T_P^o + S_u \tag{8.21}$$

where

$$a_P = a_P^o = \rho c \frac{\Delta x}{\Delta t}$$

and

Node	a_W	a_E	S_u
1	0	$k/\Delta x$	0
2, 3, 4	$k/\Delta x$	$k/\Delta x$	0
5	$k/\Delta x$	0	$\frac{2k}{\Delta x}(T_B - T_P^o)$

The time step for the explicit method is subject to the condition that

$$\Delta t < \frac{\rho c (\Delta x)^2}{2k}$$

$$\Delta t < \frac{10 \times 10^6 (0.004)^2}{2 \times 10}$$

$$\Delta t < 8 \text{ s}$$

Let us select $\Delta t = 2$ s. Substituting numerical values we have

$$\frac{k}{\Delta x} = \frac{10}{0.004} = 2500$$

$$\rho c \frac{\Delta x}{\Delta t} = 10 \times 10^6 \times \frac{0.004}{2} = 20000$$

After substitution of numerical values and some simplification the discretisation equations for the various nodes are

$$\begin{aligned} &\text{Node 1:} && 200T_P = 25T_E^o + 175T_P^o \\ &\text{Nodes 2–4:} && 200T_P = 25T_W^o + 25T_E^o + 150T_P^o \\ &\text{Node 5:} && 200T_P = 25T_W^o + 125T_P^o \end{aligned} \tag{8.22}$$

Starting with the initial condition where all the nodes are at a temperature of 200°C, the solution at each time step is obtained using equations (8.22). Although the calculations are not complicated, their number is large and they are most effectively carried out by a computer program. Table 8.1 gives a sample of the calculations for the first two time steps.

Table 8.1 Specimen calculations for the explicit method

Time	*Node 1*	*Node 2*	*Node 3*	*Node 4*	*Node 5*
$t = 0$ s	$T_1^0 = 200$	$T_2^0 = 200$	$T_3^0 = 200$	$T_4^0 = 200$	$T_5^0 = 200$
	$200T_1^1 = 25 \times 200 + 175 \times 200$	$200T_2^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_3^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_4^1 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_5^1 = 25 \times 200 + 125 \times 200$
$t = 2$ s	$T_1^1 = 200$	$T_2^1 = 200$	$T_3^1 = 200$	$T_4^1 = 200$	$T_5^1 = 150$
	$200T_1^2 = 25 \times 200 + 175 \times 200$	$200T_2^2 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_3^2 = 25 \times 200 + 25 \times 200 + 150 \times 200$	$200T_4^2 = 25 \times 200 + 25 \times 150 + 150 \times 200$	$200T_5^2 = 25 \times 200 + 125 \times 150$
$t = 4$ s	$T_1^2 = 200$	$T_2^2 = 200$	$T_3^2 = 200$	$T_4^2 = 193.75$	$T_5^2 = 118.75$

Note: Subscripts denote the node number, superscripts denote the time step

Table 8.2 shows the results for 10 consecutive time steps and Table 8.3 shows the numerical and analytical results at times 40, 80 and 120 s. As can be seen from the error analysis, the results are in good agreement with the analytical solution. Figure 8.3 shows the comparison in a graphical form.

Figure 8.4 shows the solution for time $t = 40$ s with a time step of 8 s. The previous result with a step size of 2s and the exact solution are also shown for comparison. We conclude that a time step equal to the limiting value of 8 s gives a very inaccurate and unrealistic numerical solution that oscillates about the exact solution.

Table 8.2 Results for Example 8.1 (explicit method)

		Node number						
			1	*2*	*3*	*4*	*5*	
Time step	*Time (s)*	$x = 0.0$	$x = 0.002$	$x = 0.006$	$x = 0.01$	$x = 0.014$	$x = 0.016$	$x = 0.018$
0	0	200	200	200	200	200	200	200
1	2	200	200	200	200	200	150	0
2	4	200	200	200	200	193.75	118.75	0
3	6	200	200	200	199.21	185.16	98.43	0
4	8	200	200	199.9	197.55	176.07	84.66	0
5	10	199.98	199.98	199.62	195.16	167.33	74.92	0
6	12	199.94	199.94	199.11	192.24	159.26	67.74	0
7	14	199.83	199.83	198.35	188.98	151.94	62.24	0
8	16	199.65	199.65	197.36	185.52	145.36	57.89	0
9	18	199.37	199.37	196.17	181.98	139.45	54.35	0
10	20	198.97	198.97	194.79	178.44	134.12	51.40	0

Table 8.3

	Time = 40 s			*Time = 80 s*			*Time = 120 s*		
Node	*Numerical*	*Analytical*	*% error*	*Numerical*	*Analytical*	*% error*	*Numerical*	*Analytical*	*% error*
1	188.64	188.39	–0.13	153.33	152.65	–0.43	120.53	119.87	–0.55
2	176.41	175.76	–0.36	139.05	138.36	–0.50	108.82	108.21	–0.56
3	148.29	147.13	–0.79	111.29	110.63	–0.59	86.47	85.96	–0.58
4	100.76	99.50	–1.26	72.06	71.56	–0.69	55.58	55.25	–0.60
5	35.94	35.38	–1.57	24.96	24.77	–0.75	19.16	19.05	–0.59

Figure 8.3 Comparison of numerical and analytical solutions at different times

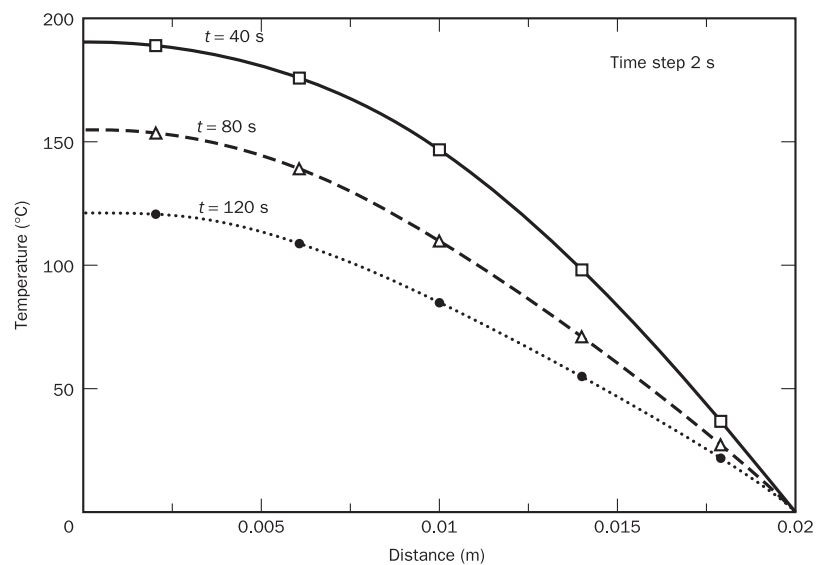
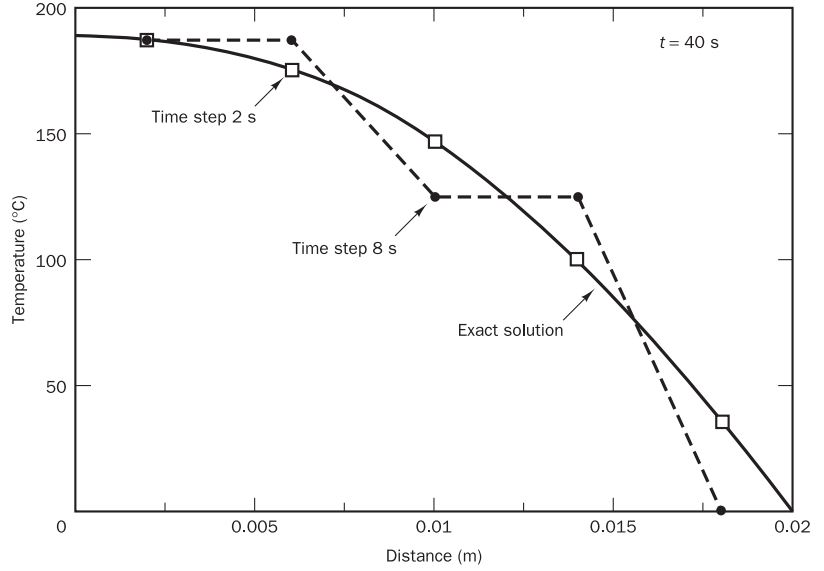



Figure 8.4 Comparison of results obtained using different time step values

Example 8.2

Solve the problem of Example 8.1 again using the fully implicit method and compare the explicit and implicit method solutions for a time step of 8 s.

Solution

Let us use the same grid arrangement as in Figure 8.2. The fully implicit method describes events at internal control volumes 2, 3 and 4 by means of discretised equation (8.16). Boundary control volumes 1 and 5 again need special treatment. Upon incorporating the boundary conditions into equation (8.9) we get for node 1

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[\frac{k}{\Delta x}(T_E - T_P) \right] - 0 \tag{8.23}$$

and for node 5

$$\rho c \frac{(T_P - T_P^o)}{\Delta t} \Delta x = \left[\frac{k}{\Delta x/2}(T_B - T_P) \right] - \left[\frac{k}{\Delta x}(T_P - T_W) \right] \tag{8.24}$$

The discretised equations are written in standard form:

$$a_P T_P = a_W T_W + a_E T_E + a_P^o T_P^o + S_u \tag{8.25}$$

where

$$a_P = a_W + a_E + a_P^o - S_P$$

and

$$a_P^o = \rho c \frac{\Delta x}{\Delta t}$$

and

Node	a_W	a_E	S_P	S_u
1	0	$k/\Delta x$	0	0
2, 3, 4	$k/\Delta x$	$k/\Delta x$	0	0
5	$k/\Delta x$	0	$-\dfrac{2k}{\Delta x}$	$\dfrac{2k}{\Delta x}T_B$

Although the implicit method permits large values for the time step Δt, we will use reasonably small time steps of 2 s to ensure good accuracy. The grid spacing and other data are as before so again we have

$$\frac{k}{\Delta x} = \frac{10}{0.004} = 2500$$

$$\rho c \frac{\Delta x}{\Delta t} = 10 \times 10^6 \times \frac{0.004}{2} = 20000$$

After substitution of numerical values and the necessary simplification the discretised equations for the various nodes are

$$\text{Node 1:} \qquad 225T_P = 25T_E + 200T_P^o$$

$$\text{Nodes 2–4:} \qquad 250T_P = 25T_W + 25T_E + 200T_P^o$$

$$\text{Node 5:} \qquad 275T_P = 25T_W + 200T_P^o + 50T_B$$

Noting that $T_B = 0$, the set of equations to be solved at each time step is

$$\begin{bmatrix} 225 & -25 & 0 & 0 & 0 \\ -25 & 250 & -25 & 0 & 0 \\ 0 & -25 & 250 & -25 & 0 \\ 0 & 0 & -25 & 250 & -25 \\ 0 & 0 & 0 & -25 & 275 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} 200T_1^o \\ 200T_2^o \\ 200T_3^o \\ 200T_4^o \\ 200T_5^o \end{bmatrix} \tag{8.26}$$

The matrix form emphasises that the equations for each point contain unknown neighbouring temperatures. The explicit scheme involves a straightforward evaluation of a single algebraic equation to find each new nodal temperature, but the fully implicit method requires the (more expensive) solution of system (8.26) at each time level. The values of temperature at the previous time level are used to calculate the right hand side. Table 8.4 and Figure 8.5 show that the numerical results again compare favourably with the analytical solution.

Table 8.4

	Time = 40 s			*Time = 80 s*			*Time = 120 s*		
Node	*Numerical*	*Analytical*	*% error*	*Numerical*	*Analytical*	*% error*	*Numerical*	*Analytical*	*% error*
1	187.38	188.38	0.51	153.72	152.65	−0.70	121.52	119.87	−1.42
2	176.28	175.76	−0.29	139.79	138.36	−1.03	109.78	108.21	−1.24
3	150.04	147.13	−1.97	112.38	110.63	−1.57	87.33	85.96	−1.59
4	103.69	99.50	−4.20	73.09	71.56	−2.13	56.20	55.25	−1.71
5	37.51	35.38	−6.02	25.38	24.77	−2.46	19.39	19.05	−1.78

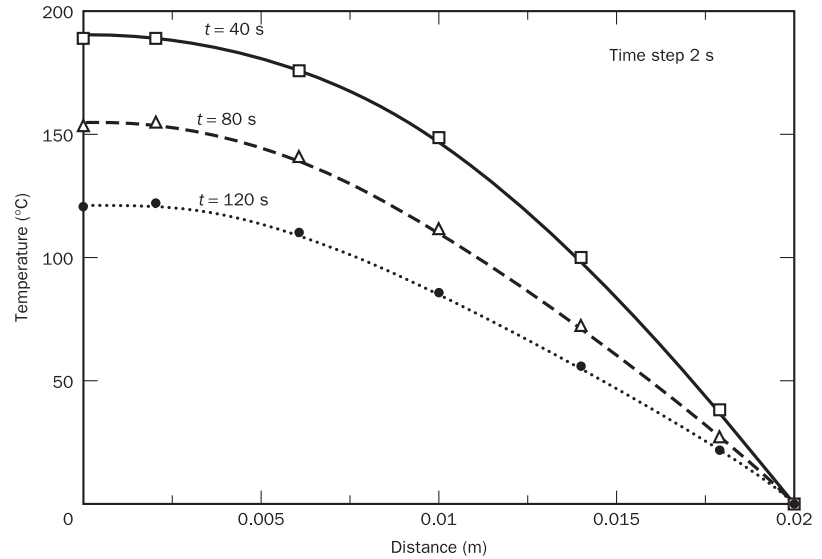
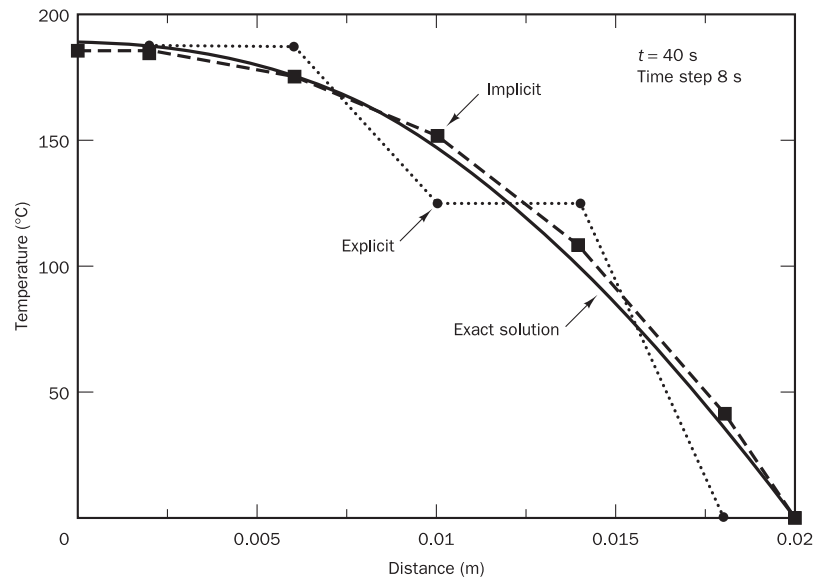
Figure 8.5 Comparison of numerical results with the analytical solution (implicit method)

In Figure 8.6 we give the solution at $t = 40$ s obtained using the implicit and explicit method with a time step of 8 s along with the analytical solution. Whereas the explicit method gives unrealistic oscillations at this step size, the implicit method gives results that are in reasonable agreement with the exact solution. This clearly illustrates a key advantage of the implicit method, which tolerates much larger time steps. However, we stress that good solution accuracy can, of course, only be achieved with small time steps.

Figure 8.6 Comparison of implicit and explicit solutions for $\Delta t = 8$ s

8.4 Implicit method for two- and three-dimensional problems

The fully implicit method is recommended for general-purpose CFD computations on the grounds of its superior stability. We now quote its extension to calculations in two and three space dimensions. Transient diffusion in three dimensions is governed by

$$\rho c\frac{\partial\phi}{\partial t}=\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial\phi}{\partial y}\right)+\frac{\partial}{\partial z}\left(\Gamma\frac{\partial\phi}{\partial z}\right)+S \tag{8.27}$$

A three-dimensional control volume is considered for the discretisation. The resulting equation is

$$\boxed{a_P\phi_P=a_W\phi_W+a_E\phi_E+a_S\phi_S+a_N\phi_N+a_B\phi_B+a_T\phi_T+a_P^o\phi_P^o+S_u} \tag{8.28}$$

where

$$\boxed{a_P=a_W+a_E+a_S+a_N+a_B+a_T+a_P^o-S_P}$$

$$\boxed{a_P^o=\rho c\frac{\Delta V}{\Delta t}}$$

The neighbouring coefficients are a_W, a_E in one-dimensional problems, and a_W, a_E, a_S, a_N in two and a_W, a_E, a_S, a_N, a_B, a_T in three dimensions; $b=(S_u+S_p\phi_P)$ is the linearised source. A summary of the relevant neighbour coefficients is given below:

	a_W	a_E	a_S	a_N	a_B	a_T
1D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	–	–	–	–
2D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	–	–
3D	$\frac{\Gamma_w A_w}{\delta x_{WP}}$	$\frac{\Gamma_e A_e}{\delta x_{PE}}$	$\frac{\Gamma_s A_s}{\delta y_{SP}}$	$\frac{\Gamma_n A_n}{\delta y_{PN}}$	$\frac{\Gamma_b A_b}{\delta z_{BP}}$	$\frac{\Gamma_t A_t}{\delta z_{PT}}$

The following values for the volume and cell face areas apply in the three cases:

	1D	*2D*	*3D*
ΔV	Δx	$\Delta x\Delta y$	$\Delta x\Delta y\Delta z$
$A_w=A_e$	1	Δy	$\Delta y\Delta z$
$A_n=A_s$	–	Δx	$\Delta x\Delta z$
$A_b=A_t$	–	–	$\Delta x\Delta y$

8.5 Discretisation of transient convection–diffusion equation

In the fully implicit discretisation approach outlined above for multi-dimensional diffusion problems, the term arising from temporal discretisation appears as (i) the contribution of a_P^o to the central coefficient a_P and (ii) the contribution of $a_P^o \phi_P^o$ as an additional source term on the right hand side. The other coefficients are unaltered and are the same as in the discretised equations for steady state problems. Using this as a basis the discretised equations for transient convection–diffusion equations are also simple to obtain. The unsteady transport of a property ϕ is given by

$$\frac{\partial}{\partial t}(\rho\phi) + \text{div}(\rho\mathbf{u}\phi) = \text{div}(\Gamma \text{ grad } \phi) + S_\phi \tag{8.29}$$

The hybrid differencing scheme was recommended in Chapter 5 on the grounds of its stability as the preferred method for treatment of convection terms, so here we quote the implicit/hybrid difference form of the transient convection–diffusion equations.

Transient three-dimensional convection–diffusion of a general property ϕ in a velocity field $\mathbf{u}$ is governed by

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u\phi)}{\partial x} + \frac{\partial(\rho v\phi)}{\partial y} + \frac{\partial(\rho w\phi)}{\partial z}$$
$$= \frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma\frac{\partial\phi}{\partial y}\right) + \frac{\partial}{\partial z}\left(\Gamma\frac{\partial\phi}{\partial z}\right) + S \tag{8.30}$$

The fully implicit discretisation equation is

$$\boxed{\begin{aligned} a_P\phi_P = {} & a_W\phi_W + a_E\phi_E + a_S\phi_S + a_N\phi_N \\ & + a_B\phi_B + a_T\phi_T + a_P^o\phi_P^o + S_u \end{aligned}} \tag{8.31}$$

where

$$\boxed{a_P = a_W + a_E + a_S + a_N + a_B + a_T + a_P^o + \Delta F - S_P}$$

with

$$\boxed{a_P^o = \frac{\rho_P^o \Delta V}{\Delta t}}$$

and

$$\boxed{\bar{S}\Delta V = S_u + S_p\phi_P}$$

The neighbour coefficients of this equation for the hybrid differencing scheme are as follows:

	1D flow	*2D flow*	*3D flow*
a_W	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$
a_E	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$
a_S	–	$\max\left[F_s, \left(D_s + \frac{F_s}{2}\right), 0\right]$	$\max\left[F_s, \left(D_s + \frac{F_s}{2}\right), 0\right]$
a_N	–	$\max\left[-F_n, \left(D_n - \frac{F_n}{2}\right), 0\right]$	$\max\left[-F_n, \left(D_n - \frac{F_n}{2}\right), 0\right]$
a_B	–	–	$\max\left[F_b, \left(D_b + \frac{F_b}{2}\right), 0\right]$
a_T	–	–	$\max\left[-F_t, \left(D_t - \frac{F_t}{2}\right), 0\right]$
ΔF	$F_e - F_w$	$F_e - F_w + F_n - F_s$	$F_e - F_w + F_n - F_s + F_t - F_b$

In the above expressions the values of F and D are calculated with the following formulae:

Face	*w*	*e*	*s*	*n*	*b*	*t*
F	$(\rho u)_w A_w$	$(\rho u)_e A_e$	$(\rho v)_s A_s$	$(\rho v)_n A_n$	$(\rho w)_b A_b$	$(\rho w)_t A_t$
D	$\frac{\Gamma_w}{\delta x_{WP}} A_w$	$\frac{\Gamma_e}{\delta x_{PE}} A_e$	$\frac{\Gamma_s}{\delta y_{SP}} A_s$	$\frac{\Gamma_n}{\delta y_{PN}} A_n$	$\frac{\Gamma_b}{\delta z_{BP}} A_b$	$\frac{\Gamma_t}{\delta z_{PT}} A_t$

The volumes and cell face areas given in section 8.4 apply here as well.

Other schemes such as linear upwind, QUICK or TVD may be incorporated into these equations by substituting the appropriate expressions for the coefficients, as will be demonstrated in the following example.

8.6 Worked example of transient convection–diffusion using QUICK differencing

Example 8.3

Consider convection and diffusion in the one-dimensional domain sketched in Figure 8.7. Calculate the transient temperature field if the initial temperature is zero everywhere and the boundary conditions are $\phi = 0$ at $x = 0$ and $\partial\phi/\partial x = 0$ at $x = L$. The data are $L = 1.5$ m, $u = 2$ m/s, $\rho = 1.0$ kg/m^3 and $\Gamma = 0.03$ kg/m.s. The source distribution defined by Figure 8.8 applies at times $t > 0$ with $a = -200$, $b = 100$, $x_1 = 0.6$ m, $x_2 = 0.2$ m. Write a computer

program to calculate the transient temperature distribution until it reaches a steady state using the implicit method for time integration and the Hayase *et al.* variant of the QUICK scheme for the convective and diffusive terms, and compare this result with the analytical steady state solution.

Figure 8.7

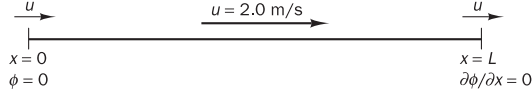
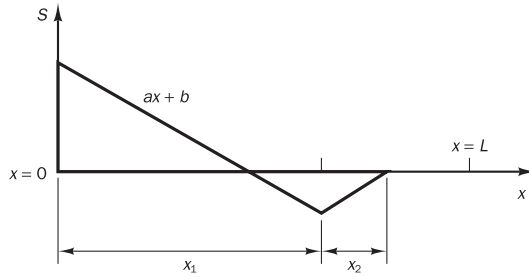



Figure 8.8 Geometry and the source distribution for Example 8.3

Solution

Transient convection–diffusion of a property ϕ subjected to a distributed source term is governed by

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u\phi)}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right) + S \tag{8.32}$$

We use a 45-point grid to sub-divide the domain and perform all calculations with a computer program. It is convenient to use the Hayase *et al.* formulation of QUICK (see section 5.9.3) since it gives a tri-diagonal system of equations which can be solved iteratively with the TDMA (see section 7.2).

The velocity is $u = 2.0$ m/s and the cell width is $\Delta x = 0.0333$ so $F = \rho u = 2.0$ and $D = \Gamma/\Delta x = 0.9$ everywhere. The Hayase *et al.* formulation gives ϕ at cell faces by means of the following formulae:

$$\phi_e = \phi_P + \frac{1}{8}(3\phi_E - 2\phi_P - \phi_W) \tag{8.33}$$

$$\phi_w = \phi_W + \frac{1}{8}(3\phi_P - 2\phi_W - \phi_{WW}) \tag{8.34}$$

The implicit discretisation equation at a general node with Hayase *et al.*'s QUICK scheme is given by

$$\frac{\rho(\phi_P - \phi_P^o)\Delta x}{\Delta t} + F_e\left[\phi_P + \frac{1}{8}(3\phi_E - 2\phi_P - \phi_W)\right] - F_w\left[\phi_W + \frac{1}{8}(3\phi_P - 2\phi_W - \phi_{WW})\right] = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) \tag{8.35}$$

The first and last nodes need to be treated separately. At control volume 1 the mirror node approach, introduced in section 5.9.1, can be used to create

a west (W) node beyond the boundary at $x = 0$. Since $\phi_A = 0$ at this boundary (A) the linearly extrapolated value at the mirror node is given by

$$\phi_0 = -\phi_P \tag{8.36}$$

and the diffusive flux at the boundary by

$$\Gamma \frac{\partial \phi}{\partial x}\bigg|_A = \frac{D_A^*}{3}(9\phi_P - 8\phi_A - \phi_E) \tag{8.37}$$

where $D_A^* = \Gamma/\Delta x$

The discretisation equation at node 1 may be written as

$$\frac{\rho(\phi_P - \phi_P^o)\Delta x}{\Delta t} + F_e\left[\phi_P + \frac{1}{8}(3\phi_E - \phi_P)\right] - F_A\phi_A$$

$$= D_e(\phi_E - \phi_P) - \frac{D_A^*}{3}(9\phi_P - 8\phi_A - \phi_E) \tag{8.38}$$

At the last control volume, the zero gradient boundary condition applies so the diffusive flux through the boundary B equals zero and the value ϕ at the boundary is equal to the upstream nodal value, i.e. $\phi_B = \phi_P$. The discretisation equation for control volume 45 becomes

$$\frac{\rho(\phi_P - \phi_P^o)\Delta x}{\Delta t} + F_B\phi_P - F_w\left[\phi_W + \frac{1}{8}(3\phi_P - 2\phi_W - \phi_{WW})\right]$$

$$= 0 - D_w(\phi_P - \phi_W) \tag{8.39}$$

These discretisation equations (8.35), (8.37) and (8.40) are now cast in standard form:

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_P^o\phi_P^o + S_u \tag{8.40}$$

with

$$a_P = a_W + a_E + a_P^o + (F_e - F_w) - S_P$$

$$a_P^o = \frac{\rho\Delta x}{\Delta t}$$

and

Node	a_W	a_E	S_P	S_u
1	0	$D_e + \frac{D_A^*}{3}$	$-\left(\frac{8}{3}D_A^* + F_A\right)$	$\left(\frac{8}{3}D_A^* + F_A\right)\phi_A + \frac{1}{8}F_e(\phi_P - 3\phi_E)$
2	$D_w + F_w$	D_e	0	$\frac{1}{8}F_w(3\phi_P - \phi_W) + \frac{1}{8}F_e(\phi_W + 2\phi_P - 3\phi_E)$
3–44	$D_w + F_w$	D_e	0	$\frac{1}{8}F_w(3\phi_P - 2\phi_W - \phi_{WW}) + \frac{1}{8}F_e(\phi_W + 2\phi_P - 3\phi_E)$
45	$D_w + F_w$	0	0	$\frac{1}{8}F_w(3\phi_P - 2\phi_W - \phi_{WW})$

The discretisation equation for control volume 2 has been adjusted to take into account the special expression that was used to evaluate the convective flux through the cell face it has in common with control volume 1.

A time step $\Delta t = 0.01$ s is selected, which is well within the stability limit for explicit schemes, so we can look forward to reasonably accurate and stable results with the implicit method. At any given time level substitution of numerical values gives the coefficients summarised in Table 8.5.

Table 8.5

Node	a_W	a_E	a_P^o	*Total source*	S_P	a_P
1	0	1.2	3.33	$4.4\phi_A + 0.25(\phi_P - 3\phi_E) + 3.33\phi_P^o$	−4.4	8.93
2	2.9	0.9	3.33	$0.25(5\phi_P - 3\phi_E) + 3.33\phi_P^o$	0	7.13
3–44	2.9	0.9	3.33	$0.25(5\phi_P - \phi_W - \phi_{WW} - 3\phi_E) + 3.33\phi_P^o$	0	7.13
45	2.9	0	3.33	$0.25(3\phi_P - 2\phi_W - \phi_{WW}) + 3.33\phi_P^o$	0	6.23

Starting with an initial field of $\phi_P^o = 0$ at all nodes, the set of equations defined by the coefficients and source contributions in Table 8.5 is solved iteratively until a converged solution ϕ_P is obtained. Subsequently, the ϕ_P-values at the current time level are assigned to ϕ_P^o and the solution proceeds to the next time level. To monitor whether the steady state has been reached we track the difference between old and new ϕ_P-values. When this attains a magnitude less than a prescribed small tolerance (say 10^{-9}) the solution is regarded as having reached the steady state.

The analytical solution

To find the exact steady state solution of (8.32) its time derivative is set to zero and the resulting ordinary differential equation is integrated twice with respect to x. The even periodic extension of the source distribution on an interval $(-L, L)$ is represented by means of a Fourier cosine series, which gives the forcing function in the differential equation. Under the given boundary conditions the solution to the problem is as follows:

$$\phi(x) = C_1 + C_2 e^{Px} - \frac{a_0}{P^2}(Px+1) - \sum_{n=1}^{\infty} a_n \left(\frac{L}{n\pi}\right)\left[P \sin\left(\frac{n\pi x}{L}\right) + \left(\frac{n\pi}{L}\right)\cos\left(\frac{n\pi x}{L}\right)\right] \Bigg/ \left[P^2 + \left(\frac{n\pi}{L}\right)^2\right] \tag{8.41}$$

with

$$P = \frac{\rho u}{\Gamma} \qquad C_2 = \frac{a_0}{P^2 e^{PL}} + \sum_{n=1}^{\infty} \frac{a_n}{e^{PL}} \cos(n\pi) \Bigg/ \left[P^2 + \left(\frac{n\pi}{L}\right)^2\right]$$

and

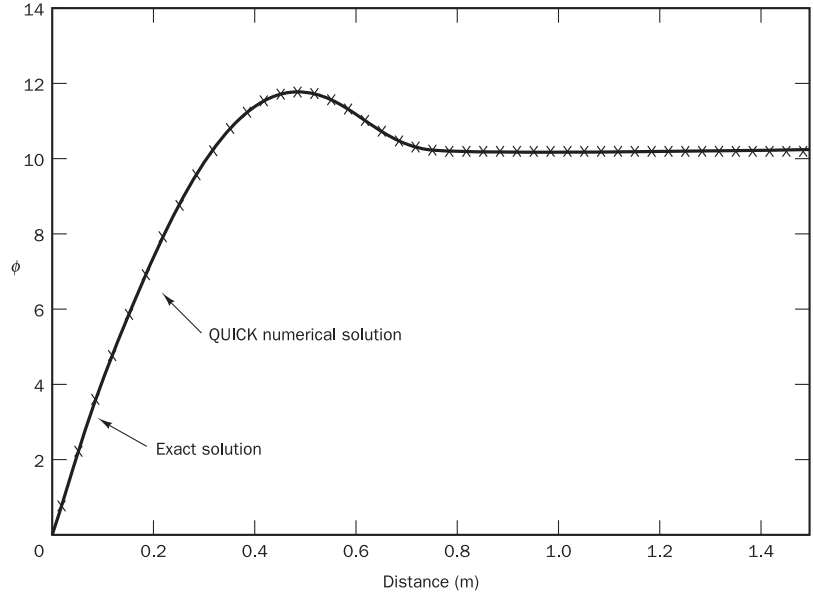
$$C_1 = -C_2 + \frac{a_0}{P^2} + \sum_{n=1}^{\infty} a_n \Bigg/ \left[P^2 + \left(\frac{n\pi}{L}\right)^2\right]$$

and

$$a_0 = \frac{(x_1 + x_2)(ax_1 + b) + bx_1}{2L}$$

$$a_n = \frac{2L}{n^2\pi^2}\left\{\left(\frac{a(x_1 + x_2) + b}{x_2}\right)\cos\left(\frac{n\pi x_1}{L}\right) - \left[a + \left(\frac{ax_1 + b}{x_2}\right)\cos\left(\frac{n\pi(x_1 + x_2)}{L}\right)\right]\right\}$$

The analytical and numerical steady state solutions are compared in Figure 8.9. As can be seen, the use of the QUICK scheme and a fine grid for spatial discretisation ensures near-perfect agreement.

Figure 8.9 Comparison of the numerical results with the analytical solution

8.7 Solution procedures for unsteady flow calculations

8.7.1 Transient SIMPLE

Algorithms such as SIMPLE, described in Chapter 6 for the calculation of steady flows, may be extended to transient calculations. The discretised momentum equations will now include transient terms formulated with the procedure described in section 8.5. An additional term is also required in the pressure correction equation. The continuity equation in a transient two-dimensional flow is given by

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \qquad (8.42)$$

The integrated form of this equation over a two-dimensional scalar control volume becomes

$$\frac{(\rho_P - \rho_P^o)}{\Delta t}\Delta V + [(\rho uA)_e - (\rho uA)_w] + [(\rho uA)_n - (\rho uA)_s] = 0 \quad (8.43)$$

The pressure correction equation is derived from the continuity equation and should therefore contain terms representing its transient behaviour. For example, the equivalent of pressure correction equation (6.32) for a two-dimensional transient flow will take the form

$$a_{I,J}p'_{I,J} = a_{I+1,J}p'_{I+1,J} + a_{I-1,J}p'_{I-1,J} + a_{I,J+1}p'_{I,J+1} + a_{I,J-1}p'_{I,J-1} + b'_{I,J} \quad (8.44)$$

where

$$a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1}$$

and

$$b'_{I,J} = (\rho u^*A)_{i,J} - (\rho u^*A)_{i+1,J} + (\rho v^*A)_{I,j} - (\rho v^*A)_{I,j+1} + \frac{(\rho_P^o - \rho_P)\Delta V}{\Delta t}$$

with neighbour coefficients

$a_{I-1,J}$	$a_{I+1,J}$	$a_{I,J-1}$	$a_{I,J+1}$
$(\rho dA)_{i,J}$	$(\rho dA)_{i+1,J}$	$(\rho dA)_{I,j}$	$(\rho dA)_{I,j+1}$

The extension to three-dimensional flows includes the same extra term in the source.

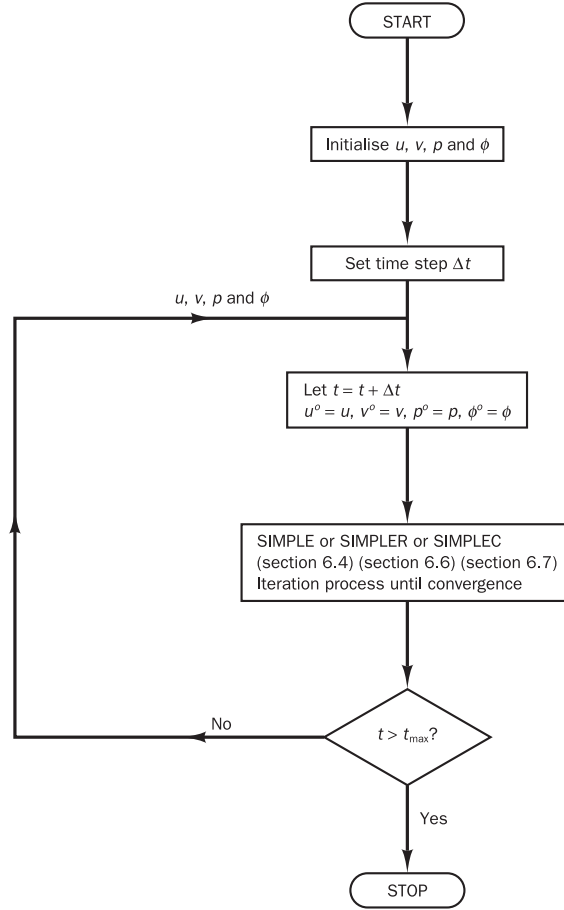
In transient flow calculations with the implicit formulation, the iterative procedures described for steady state calculations employing SIMPLE, SIMPLER or SIMPLEC are applied at each time level until convergence is achieved. Figure 8.10 shows the algorithm structure.

8.7.2 The transient PISO algorithm

The PISO algorithm is a non-iterative transient calculation procedure. It relies on the temporal accuracy gained by the discretisation practice, in particular the operator splitting technique (Issa, 1986). In the transient algorithm all time-dependent terms are retained in the momentum and continuity equations. This gives the following additional contributions to the momentum and pressure correction equations in the transient form of PISO:

- add $a_P^o = \rho_P^o \Delta V/\Delta t$ to the central coefficients of the discretised u- and v-momentum equations (6.12)–(6.13) and (6.52)–(6.53) respectively
- add $a_P^o u_P^o$ and $a_P^o v_P^o$ to the source terms of the u- and v-momentum equations
- add $(\rho_P^o - \rho_P)\Delta V/\Delta t$ to the source term of both the first and second discretised pressure correction equations

Otherwise the basic equations and steps involved in the transient version of the PISO algorithm are the same as those set out in section 6.8. The PISO procedure explained there is carried out at each time level to calculate the velocity and pressure fields. Issa (1986) shows that the temporal accuracy

Figure 8.10 Transient flow SIMPLE algorithm and its variants

achieved by the predictor–corrector process for pressure and momentum is third-order (Δt^3) and fourth-order (Δt^4) respectively. Therefore, the pressure and velocity fields obtained at the end of the PISO process with a suitably small time step are considered to be accurate enough to proceed to the next time step immediately and the algorithm is non-iterative.

Since the algorithm relies on the higher-order temporal accuracy gained by the splitting technique, small time steps are recommended to ensure accurate results. If necessary, a higher-order temporal differencing scheme may be incorporated in the algorithm for improved performance, such as a second-order implicit scheme that uses three time levels $n + 1$, n, $n - 1$ at intervals of Δt. We may use the gradient at time level n of the quadratic profile passing through T_{n+1}, T_n and T_{n-1} to evaluate $\partial T/\partial t$. The resulting time discretisation with second-order accuracy is

$$\frac{\partial T}{\partial t} = \frac{1}{2\Delta t}(3T^{n+1} - 4T^n + T^{n-1}) \tag{8.45}$$

Incorporation of the scheme to formulate discretised equations is relatively straightforward. The values at time level n and $n - 1$ known from previous time steps are treated as source terms and are placed on the right hand side of the equation.

The PISO method has yielded accurate results with sufficiently small time steps (see e.g. Issa *et al*, 1986; Kim and Benson, 1992). Since the PISO method does not require iterations within a time level it is less expensive than the implicit SIMPLE algorithm. CFD simulation of flow and heat transfer in internal combustion engines requires transient calculations that are inevitably time consuming and expensive, especially with three-dimensional geometries. Ahmadi-Befrui *et al* (1990) have presented a version of PISO known as EPISO suitable for predicting engine flows.

8.8 Steady state calculations using the pseudo-transient approach

It was mentioned in Chapter 6 that under-relaxation is necessary to stabilise the iterative process of obtaining steady state solutions. The under-relaxed form of the two-dimensional u-momentum equation, for example, takes the form

$$\frac{a_{i,J}}{\alpha_u} u_{i,J} = \sum a_{nb} u_{nb} + (p_{I-1,J} - p_{I,J}) A_{i,J} + b_{i,J} + \left[(1-\alpha_u) \frac{a_{i,J}}{\alpha_u} \right] u_{i,J}^{(n-1)} \tag{8.46}$$

Compare this with the transient (implicit) u-momentum equation

$$\left(a_{i,J} + \frac{\rho_{i,J}^{o} \Delta V}{\Delta t} \right) u_{i,J} = \sum a_{nb} u_{nb} + (p_{I-1,J} - p_{I,J}) A_{i,J} + b_{i,J} + \frac{\rho_{i,J}^{o} \Delta V}{\Delta t} u_{i,J}^{o} \tag{8.47}$$

In equation (8.46) the superscript $(n - 1)$ indicates the previous iteration and in equation (8.47) superscript o represents the previous time level. We immediately note a clear analogy between transient calculations and under-relaxation in steady state calculations. It can be easily deduced that

$$(1-\alpha_u) \frac{a_{i,J}}{\alpha_u} = \frac{\rho_{i,J}^{o} \Delta V}{\Delta t} \tag{8.48}$$

This formula shows that it is possible to achieve the effects of under-relaxed iterative steady state calculations from a given initial field by means of a pseudo-transient computation starting from the same initial field by taking a step size that satisfies (8.48). Alternatively steady state calculations may be interpreted as pseudo-transient solutions with spatially varying time steps. The pseudo-transient approach is useful for situations in which governing equations give rise to stability problems, e.g. buoyant flows, highly swirling flows and compressible flows with shocks.

8.9 A brief note on other transient schemes

Other transient flow calculation procedures such as MAC (Harlow and Welch, 1965), SMAC (Amsden and Harlow, 1970), ICE (Harlow and Amsden, 1971) and ICED-ALE (Hirt *et al*., 1974) are available to the user. The calculation methodology of this class of schemes includes the direct solution of a Poisson equation for the pressure as a central feature of the algorithm. The overall calculation process is different from the techniques explained here and the interested reader is referred to cited references for more details. The well-known engine prediction code KIVA uses the ICED-ALE method as the core solution procedure. The method has been shown to be reliable for predicting practical internal combustion engine flows and is widely used for internal combustion engine research (see Amsden *et al*., 1985, 1989; Zellat

et al., 1990; Blunsdon *et al.*, 1992, 1993). Kim and Benson (1992) compared the PISO method with the SMAC algorithms for the prediction of unsteady flows and reported that SMAC was more efficient, faster and more accurate than PISO. The MAC/ICE class of methods are, however, mathematically complex and not widely used in general-purpose CFD procedures.

8.10 Summary

Techniques for the solution of transient flow problems were developed by considering the unsteady diffusion and convection–diffusion equations. We distinguish between the following time-stepping algorithms for the computation of a variable ϕ at a new time level:

- explicit – uses only ϕ from the previous time level
- Crank–Nicolson – uses a mixture of ϕ from the previous time level and ϕ at a new time level
- implicit – uses mainly surrounding ϕ-values at the new time level

The stability and accuracy properties of each of the schemes are given in Table 8.6 and described below.

Table 8.6

Scheme	*Stability*	*Accuracy*	*Positive coefficient criterion*
Explicit	Conditionally stable	First-order	$\Delta t < \dfrac{\rho(\Delta x)^2}{2\Gamma}$
Crank–Nicolson	Unconditionally stable	Second-order	$\Delta t < \dfrac{\rho(\Delta x)^2}{\Gamma}$
Implicit	Unconditionally stable	First-order	Always positive

- For robust general-purpose transient CFD calculations the implicit scheme is recommended. The unconditional stability of this and the Crank–Nicolson scheme is, however, bought at the price of having to solve a system of equations at each time level. In two- and three-dimensional calculations this requires intermediate iterative stages.
- The (fully implicit) transient discretisation equations for diffusion and convection–diffusion are practically the same as those of steady problems apart from minor changes to the central coefficient a_P and the source term b_P:

  $$a_P^{(t)} = a_P^{(s)} + a_P^o \text{ and } b_P^{(t)} = b_P^{(s)} + a_P^o \phi_P^o \text{ with } a_P^o = \rho_P^o \Delta V/\Delta t$$

  The superscript (t) refers to the transient form and (s) to the steady form.
- In addition to the above modifications to the momentum equations in SIMPLE its pressure correction equation also requires an addition of $(\rho_P^o - \rho_P)\Delta V/\Delta t$ to the source term b_P. The time-stepping procedure creates an extra loop outside the main iteration cycles of SIMPLE.
- The time accuracy of the second corrector step of PISO makes it very attractive for non-iterative transient calculations.
- The similarity between the under-relaxed iterative solution and the pseudo-transient solution was highlighted. The pseudo-transient strategy has been widely used to combat stability problems in flows with complex physics.

Chapter nine Implementation of boundary conditions

9.1 Introduction

All CFD problems are defined in terms of initial and boundary conditions. It is important that the user specifies these correctly and understands their role in the numerical algorithm. In transient problems the initial values of all the flow variables need to be specified at all solution points in the flow domain. Since this involves no special measures other than initialising the appropriate data arrays in the CFD code, we do not need to discuss this topic further. The present chapter describes the implementation in the discretised equations of the finite volume method of the most common boundary conditions:

- inlet
- outlet
- wall
- prescribed pressure
- symmetry
- periodicity (or cyclic boundary condition)

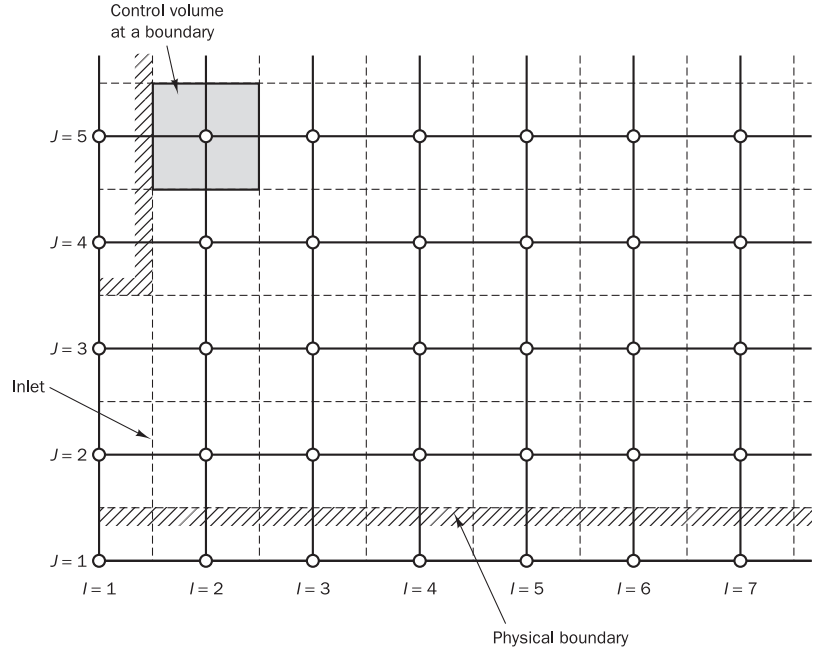
In constructing a staggered grid arrangement we set up additional nodes surrounding the physical boundary, as illustrated in Figure 9.1. The calculations are performed at internal nodes only ($I = 2$ and $J = 2$ onwards). Two notable features of the arrangement are (i) the physical boundaries coincide with scalar control volume boundaries and (ii) the nodes just outside the inlet of the domain (along $I = 1$ in Figure 9.1) are available to store the inlet conditions. This enables the introduction of boundary conditions to be achieved with small modifications to the discretised equations for near-boundary internal nodes.

In Chapters 4 and 5 we saw that boundary conditions enter the discretised equations by suppression of the link to the boundary side and modification of the source terms. The appropriate coefficient of the discretised equation is set to zero and the boundary side flux – exact or linearly approximated – is introduced through source terms S_u and S_p. We will frequently make use of this device to fix the flux of a variable at a cell face, but we also need a technique to cope with situations where we need to set the value of a variable at a node. This can be done by introducing two overwhelmingly large source terms into the relevant discretised equation. For example, to set the variable ϕ at node P to a value ϕ_{fix} the following source term modification is used in its discretised equation:

$$S_p = -10^{30} \quad \text{and} \quad S_u = 10^{30}\,\phi_{fix} \tag{9.1}$$

With these sources added to the discretised equation we have

$$(a_P + 10^{30})\phi_P = \sum a_{nb}\phi_{nb} + 10^{30}\phi_{fix} \tag{9.2}$$

Figure 9.1 The grid arrangement at boundaries

The actual magnitude of the number 10^{30} is arbitrary as long as it is very large compared with all coefficients in the original discretised equation. Thus if a_P and a_{nb} are all negligible the discretised equation effectively states that

$$\phi_P = \phi_{fix} \tag{9.3}$$

which fixes the value of ϕ at P.

In addition to setting the value of a variable at internal nodes this treatment is also useful for dealing with solid obstacles within a domain by taking $\phi_{fix} = 0$ (or any other desired value) at nodes within a solid region. The system of discretised flow equations can be solved as normal without having to deal with the obstacles separately.

Details of the modifications needed to implement the listed boundary conditions will be further explained in the text to follow. We make the following assumptions: (i) the flow is always subsonic ($M < 1$), (ii) k–ε turbulence modelling is used, (iii) the hybrid differencing method is used for discretisation and (iv) the SIMPLE solution algorithm is applied.

9.2 Inlet boundary conditions

The distribution of all flow variables needs to be specified at inlet boundaries. Here we discuss the case of an inlet perpendicular to the x-direction. Figures 9.2 to 9.5 show the grid arrangement in the immediate vicinity of an inlet for u- and v-momentum, scalar and pressure correction equation cells. The flow direction is assumed to be broadly from the left to the right in the diagrams. As mentioned, the grid extends outside the physical boundary and the nodes along the line $I = 1$ (or $i = 2$ for u-velocity) are used to store the inlet values of flow variables (indicated by u_{in}, v_{in}, ϕ_{in} and p'_{in}). Just downstream of

Figure 9.2 *u*-velocity cell at the inlet boundary

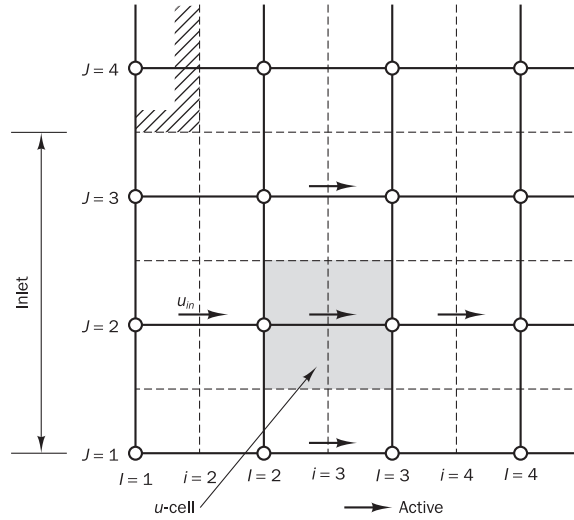
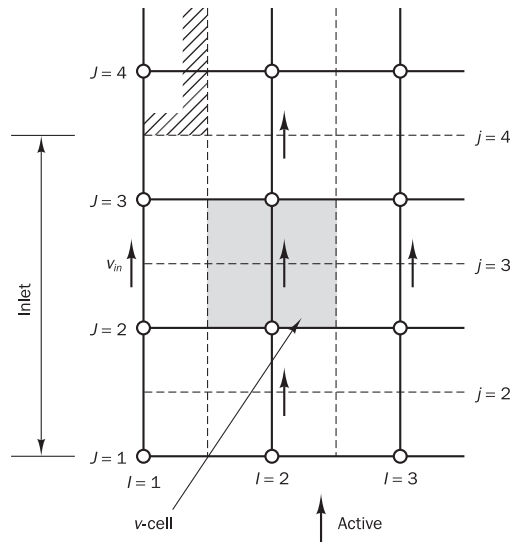



Figure 9.3 *v*-velocity cell at the inlet boundary

this extra node we start to solve the discretised equation for the first internal cell, which is shaded.

The diagrams also show the 'active' neighbours and cell faces which are represented in the discretised equation for the shaded cell assuming that hybrid differencing is used. For instance, in Figure 9.2 the active neighbour velocities are given by means of arrows and the active face pressures by open circles. The figures indicate that all links to neighbouring nodes remain active for the first u-, v- and ϕ-cell, so to accommodate the inlet boundary condition for these variables it is unnecessary to make any modifications to their discretised equations. Figure 9.4 shows that the link with the boundary side is cut in the discretised pressure correction equation by setting the boundary side (west) coefficient a_W equal to zero. Since the velocity is known

Figure 9.4 Pressure correction cell at the inlet boundary

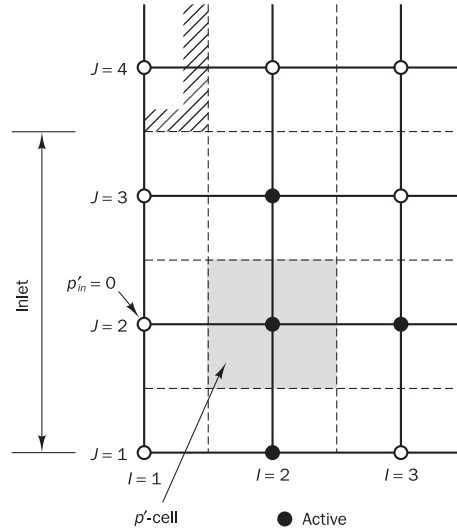
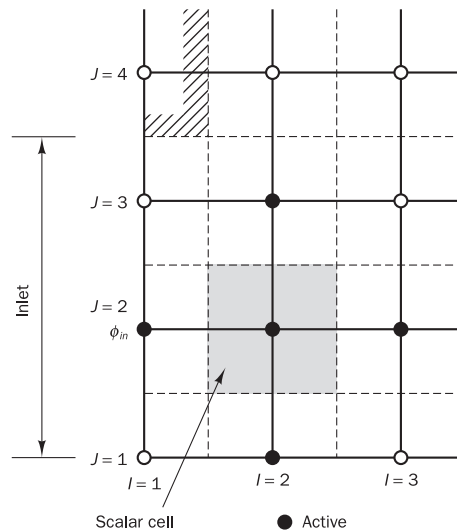



Figure 9.5 Scalar cell at the inlet boundary

at inlet, it is also not necessary to make a velocity correction here and hence we have

$$u_W^* = u_W \tag{9.4}$$

in the source associated with discretised pressure correction (6.32).

Reference pressure

The pressure field obtained by solving the pressure correction equation does not give absolute pressures (Patankar, 1980). It is common practice to fix the absolute pressure at one inlet node and set the pressure correction to zero at that node. Having specified a reference value, the absolute pressure field inside the domain can now be obtained.

Estimation of k and ε at inlet boundaries

The most accurate simulations can only be achieved by supplying measured inlet values of turbulent kinetic energy k and dissipation rate ε. However, if we perform outline design calculations such data are often not available. In this case commercial CFD codes often estimate k and ε with the approximate formulae described in section 3.7.2, based on a turbulence intensity – typically between 1% and 6% – and a length scale.

Inlet boundaries perpendicular to the y-direction

The above procedure is, of course, not restricted to an inlet boundary perpendicular to the x-direction. When we have an inlet perpendicular to the y-direction the velocity component v, for which inlet value v_{in} is available at $j = 2$, takes the place of velocity component u and the calculations start at $j = 3$. The inlet values of the remaining variables are stored at $J = 1$ and solution starts at $J = 2$. They are otherwise treated as above.

9.3 Outlet boundary conditions

Outlet boundary conditions may be used in conjunction with the inlet boundary conditions of section 9.2. If the location of the outlet is selected far away from geometrical disturbances the flow eventually reaches a fully developed state where no change occurs in the flow direction. In such a region we can place an outlet surface and state that the gradients of all variables (except pressure) are zero in the flow direction. It is normally possible to make a reasonably accurate prediction of the flow direction far away from obstacles. This gives us the opportunity to locate the outlet surface perpendicular to the flow direction and take gradients in the direction normal to the outlet surface equal to zero.

Figures 9.6 to 9.9 show grid arrangements near such an outlet boundary. We have shaded the last cells upstream of the outlet, for which a discretised equation is solved, and, as before, highlighted the active neighbours and faces.

Figure 9.6 u-control volume at an outlet boundary

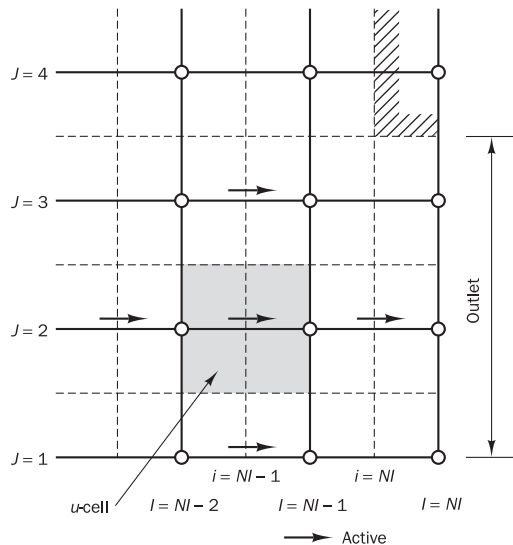



Figure 9.7 v-control volume at an outlet boundary

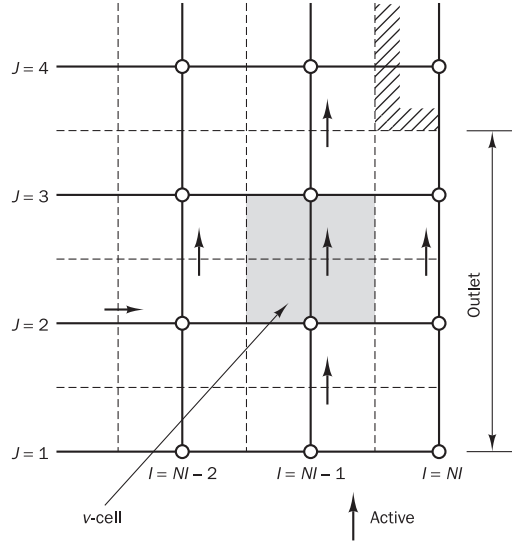
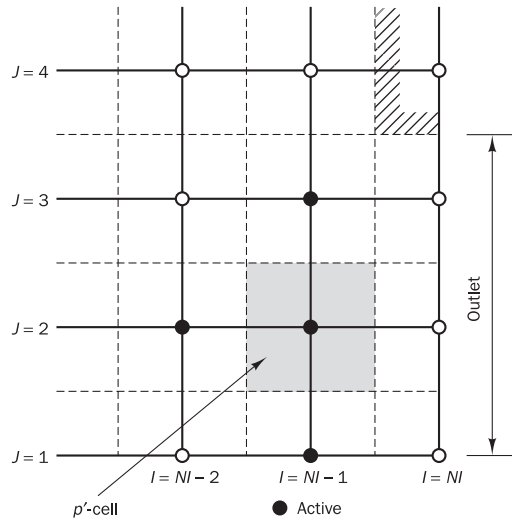
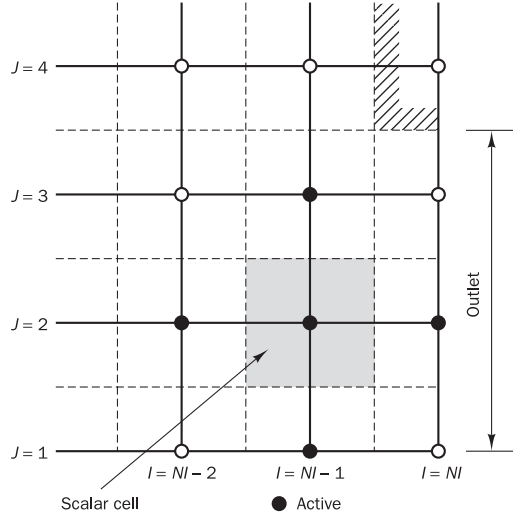



Figure 9.8 p'-control volume at an outlet boundary

If NI is the total number of nodes in the x-direction, equations are solved for cells up to I (or i) = $NI - 1$. Before the relevant equations are solved the values of flow variables at the next node (NI), just outside the domain, are determined by extrapolation from the interior on the assumption of zero gradient at the outlet plane. For the v- and scalar equations this implies setting $v_{NI,j} = v_{NI-1,j}$ and $\phi_{NI,J} = \phi_{NI-1,J}$. Figures 9.7 and 9.9 show that all links are active for these variables so their discretised equations can be solved as normal.

Special care should be taken in the case of the u-velocity. Calculation of u at the outlet plane $i = NI$ by assuming a zero gradient gives

$$u_{NI,J} = u_{NI-1,J} \tag{9.5}$$

During the iteration cycles of the SIMPLE algorithm there is no guarantee that these velocities will conserve mass over the computational domain as a

Figure 9.9 Scalar cell at an outlet boundary

whole. To ensure that overall continuity is satisfied the total mass flux going out of the domain (M_{out}) is first computed by summing all the extrapolated outlet velocities (9.5). To make the mass flux out equal to the mass flux M_{in} coming into the domain all the outlet velocity components $u_{NI,J}$ of (9.5) are multiplied by the ratio M_{in}/M_{out}. Thus the outlet plane velocities with the continuity correction are given by

$$u_{NI,J} = u_{NI-1,J} \times \frac{M_{in}}{M_{out}} \tag{9.6}$$

These values are subsequently used as the east neighbour velocities in the discretised momentum equations for $u_{NI-1,J}$.

The velocity at the outlet boundaries is not corrected by means of pressure corrections. Hence in the discretised p'-equation (6.32) the link to the outlet boundary side (east) is suppressed by setting $a_E = 0$. The contribution to the source term in this equation is calculated as normal, noting that $u_E^* = u_E$; no additional modifications are required.

9.4 Wall boundary conditions

The wall is the most common boundary encountered in confined fluid flow problems. In this section we consider a solid wall parallel to the x-direction. Figures 9.10 to 9.12 illustrate the grid details in the near-wall regions for the u-velocity component (parallel to the wall), for the v-velocity component (perpendicular to the wall) and for scalar variables.

The no-slip condition ($u = v = 0$) is the appropriate condition for the velocity components at solid walls. The normal component of the velocity can simply be set to zero at the boundary ($j = 2$), and the discretised momentum equation at the next v-cell in the flow ($j = 3$) can be evaluated without modification. Since the wall velocity is known it is also unnecessary to perform a pressure correction here. In the discretised p'-equation (6.32) for the cell nearest to the wall the wall link (south) is, therefore, cut by setting $a_S = 0$, and we take $v_s^* = v_s$ in its source term.

Figure 9.10 u-velocity cell at a wall boundary

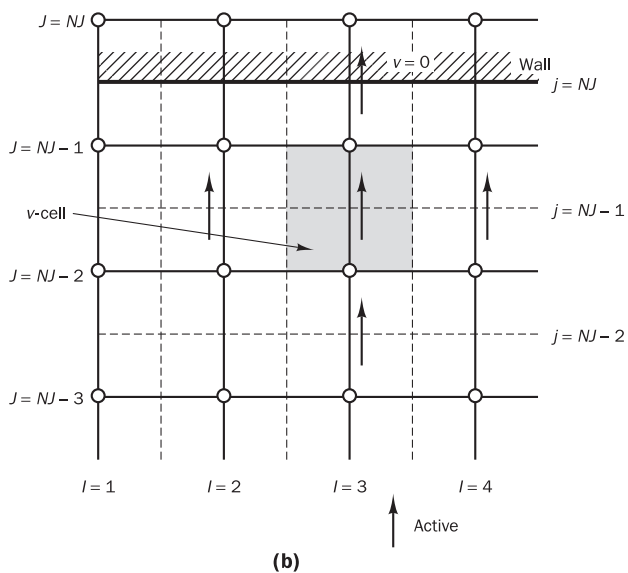
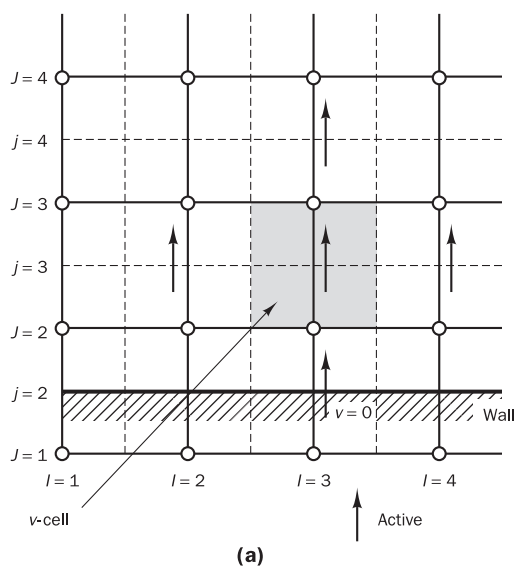
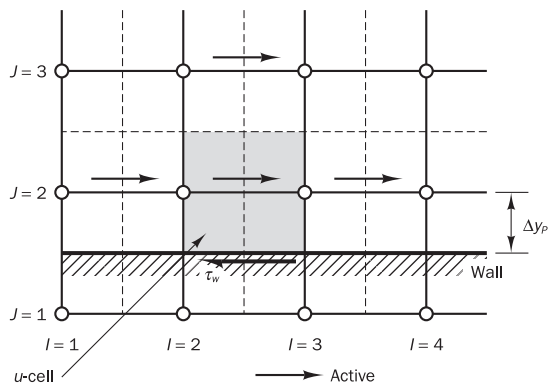
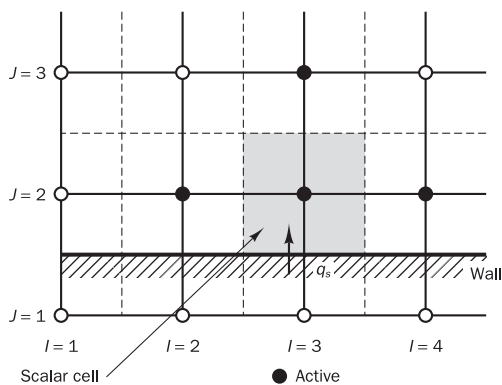



Figure 9.11 v-cell at a wall boundary: (a) $j = 3$ and (b) $j = NJ$

Figure 9.12 Scalar cell at a wall boundary

For all other variables special sources are constructed, the precise form of which depends on whether the flow is laminar or turbulent. In Chapter 3 we studied the multi-layered structure of the near-wall turbulent boundary layer. Immediately adjacent to the wall we have an extremely thin viscous sub-layer followed by the buffer layer and the turbulent core. The number of mesh points required to resolve all the details in a turbulent boundary layer would be prohibitively large, and normally we employ the 'wall functions' introduced in Chapter 3 to represent the effect of the wall boundaries.

The implementation of wall boundary conditions in turbulent flows starts with the evaluation of

$$y^+ = \frac{\Delta y_P}{\nu}\sqrt{\frac{\tau_w}{\rho}} \tag{9.7}$$

where Δy_P is the distance of the near-wall node P to the solid surface (see Figure 9.10). A near-wall flow is taken to be laminar if $y^+ \le 11.63$. The wall shear stress is assumed to be entirely viscous in origin. If $y^+ > 11.63$ the flow is turbulent and the wall function approach is used. The criterion places the changeover from laminar to turbulent near-wall flow in the buffer layer between the linear and log-law regions of a turbulent wall layer. The exact value of $y^+ = 11.63$ is the intersection of the linear profile and the log-law, so it is obtained from the solution of

$$y^+ = \frac{1}{\kappa}\ln(Ey^+) \tag{9.8}$$

In this formula κ is von Karman's constant (0.4187) and E is an integration constant that depends on the roughness of the wall (see section 3.4.2). For smooth walls with constant shear stress E has a value of 9.793.

Laminar flow/linear sub-layer

The wall conditions described under this heading apply in two cases: for solutions of (i) laminar flow equations and (ii) turbulent flow equations when $y^+ \le 11.63$. In both cases the near-wall flow is taken to be laminar. The wall force is entered into the discretised u-momentum equation as a source. The wall shear stress value is obtained from

$$\tau_w = \mu\frac{u_P}{\Delta y_P} \tag{9.9}$$

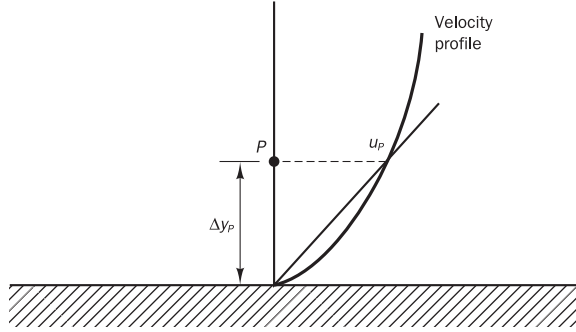
where u_P is the velocity at the grid node. Figure 9.13 illustrates that this formula is based on the assumption that the velocity varies linearly with distance from the wall in a laminar flow.

The shear force F_s is now given by

$$\begin{aligned} F_s &= -\tau_w A_{Cell} \\ &= -\mu\frac{u_P}{\Delta y_P}A_{Cell} \end{aligned} \tag{9.10}$$

where A_{Cell} is the wall area of the control volume. The appropriate source term in the u-equation is defined by

$$S_P = -\frac{\mu}{\Delta y_P}A_{Cell} \tag{9.11}$$

Figure 9.13 Velocity distribution at a wall

Heat transfer from a wall at fixed temperature T_w into the near-wall cell in laminar flow is calculated from

$$q_s = -\frac{\mu}{\sigma}\frac{C_P(T_P - T_w)}{\Delta y_P}A_{Cell} \tag{9.12}$$

where C_P is the specific heat of the fluid, T_P is the temperature at the node P and σ is the laminar Prandtl number. It is easy to see that the corresponding source terms for the temperature equation are given by

$$S_P = -\frac{\mu}{\sigma}\frac{C_P}{\Delta y_P}A_{Cell} \quad \text{and} \quad S_u = \frac{\mu}{\sigma}\frac{C_P T_w}{\Delta y_P}A_{Cell} \tag{9.13}$$

A fixed heat flux enters the source terms directly by means of the normal source term linearisation:

$$q_s = S_u + S_p T_P \tag{9.14}$$

For an adiabatic wall we have, of course, $S_u = S_p = 0$.

Turbulent flow

If the value of y^+ is greater than 11.63 node P is considered to be in the log-law region of a turbulent boundary layer. In this region wall function formulae (3.49) and (3.50) associated with the log-law are used to calculate shear stress, heat flux and other variables. The formulae have been applied in many different ways but Table 9.1 gives the optimum near-wall relationships from extensive computing trials.

These relationships should be used in conjunction with the universal velocity and temperature distributions for near-wall turbulent flows in (3.49)–(3.50):

$$u^+ = \frac{1}{\kappa}\ln(Ey^+) \tag{3.49}$$

and

$$T^+ = \sigma_{T,t}\left(u^+ + P\left[\frac{\sigma_{T,l}}{\sigma_{T,t}}\right]\right) \tag{3.50}$$

Table 9.1 Near-wall relationships for the standard k–ε model

- *Momentum equation tangential to wall*

  wall shear stress $\tau_w = \rho C_\mu^{1/4} k_P^{1/2} u_P / u^+$ (9.15)

  wall force $F_s = -\tau_w A_{Cell} = -(\rho C_\mu^{1/4} k_P^{1/2} u_P / u^+) A_{Cell}$ (9.16)

- *Momentum equation normal to wall*

  normal velocity = 0

- *Turbulent kinetic energy equation*

  net k-source per unit volume $= (\tau_w u_P - \rho C_\mu^{3/4} k_P^{3/2} u^+) \Delta V / \Delta y_P$ (9.17)

- *Dissipation rate equation*

  set nodal value $\varepsilon_P = C_\mu^{3/4} k_P^{3/2} / (\kappa \Delta y_P)$ (9.18)

- *Temperature (or energy) equation*

  wall heat flux $q_w = -C_P \rho C_\mu^{1/4} k_P^{1/2} (T_P - T_w) / T^+$ (9.19)

In these equations the values of κ and E are as given in (9.8), $\sigma_{T,l}$ is the laminar (or molecular Prandtl number, $\sigma_{T,t}$ is the turbulent Prandtl number (≈ 0.9), and function $P(\sigma_{T,l}/\sigma_{T,t})$ is called the 'pee-function', which can be evaluated using the following expression derived by Jayatilleke (1969):

$$P\left(\frac{\sigma_{T,l}}{\sigma_{T,t}}\right) = 9.24\left[\left(\frac{\sigma_{T,l}}{\sigma_{T,t}}\right)^{0.75} - 1\right] \times \left\{1 + 0.28 \exp\left[-0.007\left(\frac{\sigma_{T,l}}{\sigma_{T,t}}\right)\right]\right\} \tag{9.20}$$

In order of their appearance in Table 9.1 variables are treated as follows in their discretised equations:

- *u-velocity component parallel to the wall*. The link with the wall (south) is suppressed by setting $a_S = 0$, and wall force F_s from (9.16) is introduced into the discretised u-equation as a source term, so

$$S_P = -\frac{\rho C_\mu^{1/4} k_P^{1/2}}{u^+} A_{Cell} \tag{9.21}$$

- *k-equation*. The link at the boundary is suppressed; we set $a_S = 0$. In the volume source (9.17) the second term contains $k^{3/2}$. This is linearised as $k_P^{*1/2} \, . \, k_P$, where k^* is the k-value at the end of the previous iteration, which yields the following source terms S_p and S_u in the discretised k-equation:

$$S_p = -\frac{\rho C_\mu^{3/4} k_P^{*1/2} u^+}{\Delta y_P} \Delta V \quad \text{and} \quad S_u = \frac{\tau_w u_p}{\Delta y_P} \Delta V \tag{9.22}$$

- *ε-equation*. In the discretised ε-equation the near-wall node is fixed to the value given by (9.18) by means of setting the source terms S_p and S_u as follows:

$$S_p = -10^{30} \quad \text{and} \quad S_u = \frac{C_\mu^{3/4} k_P^{3/2}}{\kappa \Delta y_P} \times 10^{30} \tag{9.23}$$

- *Temperature equation*. The link with the wall is suppressed in the T-equation by setting the boundary side coefficient a_S to zero. The wall heat flux is calculated using equation (9.19) and introduced by means of the following source terms:

$$S_P = -\frac{\rho C_\mu^{1/4} k_P^{1/2} C_P}{T^+} A_{Cell} \quad \text{and} \quad S_u = \frac{\rho C_\mu^{1/4} k_P^{1/2} C_P T_{wall}}{T^+} A_{Cell} \tag{9.24}$$

A fixed heat flux enters the source terms directly by means of the normal source term linearisation:

$$q_s = S_u + S_p T_P \tag{9.25}$$

For an adiabatic wall we have $S_u = S_p = 0$, as before.

Rough walls

In the wall function approach described above, changeover from laminar to turbulent flow as the distance from the wall increases was assumed to occur at $y^+ = 11.63$, which is the solution of equation (9.8) with $E = 9.8$. This criterion applies to smooth walls; if walls are not smooth E should be adjusted accordingly and a new limiting value of y^+ would result. E may be estimated on the basis of measured absolute roughness values. Schlichting (1979), among others, gives further details.

Moving walls

Note that it has been tacitly assumed that the wall is stationary. Wall movement in the x-direction is felt by the fluid by a change in the wall shear stress. Its value is adjusted by replacing velocity u_P by the relative velocity $u_P - u_{wall}$. This modifies the laminar wall force formula (9.10) as follows:

$$F_s = -\mu \frac{(u_P - u_{wall})}{\Delta y_P} A_{Cell} \tag{9.26}$$

and the turbulent wall force formula (9.16) as

$$F_s = -\frac{\rho C_\mu^{1/4} k_P^{1/2} (u_P - u_{wall})}{u^+} A_{Cell} \tag{9.27}$$

The relevant source terms (9.11) and (9.21) are similarly adjusted.

Wall motion also alters the volume source term of the k-equations, which becomes

$$[\tau_w (u_P - u_{wall}) - \rho C_\mu^{3/4} k_P^{3/2} u^+] \Delta V / \Delta y_P \tag{9.28}$$

It should be noted that the wall functions described above have been derived on the basis of the following assumptions:

- the velocity is parallel to the wall and varies only in the direction normal to the wall
- no pressure gradients in the flow direction
- no chemical reactions at the wall
- high Reynolds number

If any one of these assumptions does not hold, the accuracy of the predictions using this wall function approach may be reduced or even seriously compromised.

9.5 The constant pressure boundary condition

The constant pressure condition is used in situations where exact details of the flow distribution are unknown but the boundary values of pressure are known. Typical problems where this boundary condition is appropriate include external flows around objects, free surface flows, buoyancy-driven flows such as natural ventilation and fires, and also internal flows with multiple outlets.

In applying the fixed pressure boundary the pressure correction is set to zero at the nodes. The grid arrangement of the p'-cells near a flow inlet and outlet is shown in Figures 9.14 and 9.15.

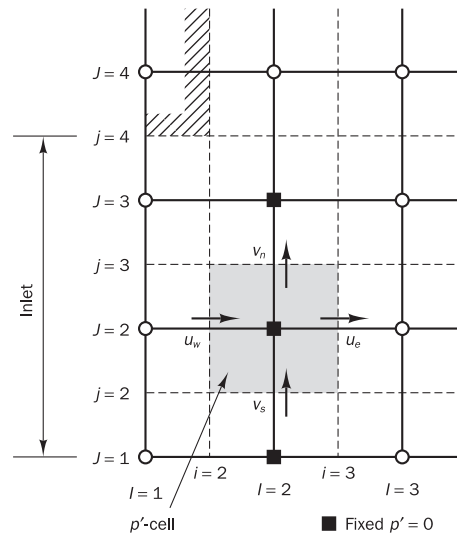
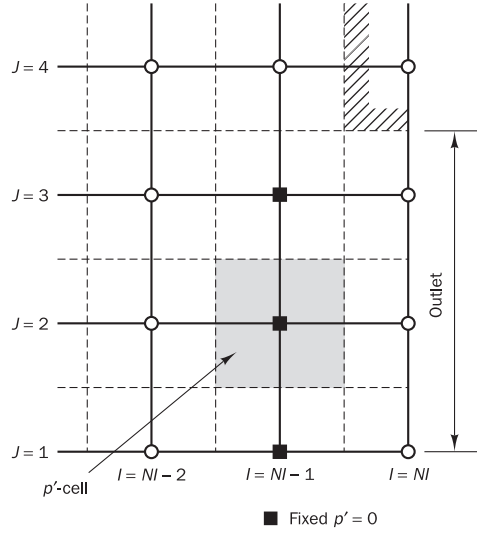
Figure 9.14 p'-cell at an inlet boundary

A convenient way of dealing with a constant pressure boundary condition is to fix pressure at the nodes just inside the physical boundary, as indicated in the diagrams by solid squares. The pressure corrections are set to zero by taking $S_u = 0.0$ and $S_p = -10^{30}$, and the nodal pressure is set to the required boundary pressure p_{fix}. The u-momentum equation is solved from $i = 3$ and v-momentum and other equations from $I = 2$ onwards. The main outstanding problem is the unknown flow direction, which is governed by the conditions inside the calculation domain. The u-velocity component across the domain boundary is generated as part of the solution process by ensuring that continuity is satisfied at every cell. For example, in Figure 9.14 the values of u_e

Figure 9.15 p'-cell at an outlet boundary

and of v_s and v_n emerge from solving the discretised u- and v-momentum equations inside the domain. Given these values we can compute u_w by insisting that mass is conserved for the p'-cell. This yields

$$u_w = \frac{(\rho vA)_n - (\rho vA)_s + (\rho uA)_e}{(\rho A)_w} \tag{9.29}$$

This implementation of the boundary condition causes the p'-cell nearest to the boundaries to act as a source or sink of mass. The process is repeated for each pressure boundary cell. Other variables such as v, T, k and ε must be assigned inflow values where the flow direction is *into* the domain. Where the flow is *outwards* their values just outside the domain may be obtained by means of extrapolation (see section 9.3).

There are several variations that can be useful in practical circumstances. Some codes apply (i) a condition at inlet that fixes the stagnation pressure of the inlet flow just outside the domain (at $i = 1$) instead of the static pressure just inside the domain (at $i = 2$) and/or (ii) the extrapolation procedure at outlets for all variables including u.

9.6 Symmetry boundary condition

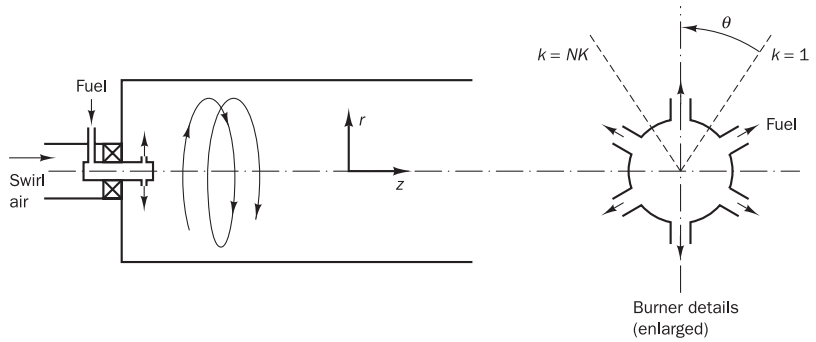
The conditions at a symmetry boundary are: (i) no flow across the boundary and (ii) no scalar flux across the boundary. In the implementation, normal velocities are set to zero at a symmetry boundary, and the values of all other properties just outside the solution domain (say I or $i = 1$) are equated to their values at the nearest node just inside the domain (I or $i = 2$):

$$\phi_{1,J} = \phi_{2,J} \tag{9.30}$$

In the discretised p'-equations the link with the symmetry boundary side is cut by setting the appropriate coefficient to zero; no further modifications are required.

9.7 Periodic or cyclic boundary condition

Periodic or cyclic boundary conditions arise due to a different type of symmetry in a problem. Consider for example swirling flow in the cylindrical furnace shown in Figure 9.16. In the burner arrangement gaseous fuel is introduced through six symmetrically placed holes and swirl air enters through the outer annulus of the burner.

Figure 9.16 An example of a cyclic boundary condition

This problem can be solved in cylindrical polar co-ordinates (z, r, θ) by considering a 60° angular sector as shown in the diagram, where k refers to r–z planes in the θ-direction. The flow rotates in this direction, and under the given conditions the flow entering the first k-plane of the sector should be exactly the same as that leaving the last k-plane. This is an example of cyclic symmetry. The pair of boundaries $k = 1$ and $k = NK$ are called periodic or cyclic boundaries.

To apply cyclic boundary conditions we need to set the flux of all flow variables leaving the outlet cyclic boundary equal to the flux entering the inlet cyclic boundary. This is achieved by equating the values of each variable at the nodes just upstream and downstream of the inlet plane to the nodal values just upstream and downstream of the outlet plane. For all variables except the velocity component across the inlet and outlet planes (say w) we have

$$\phi_{1,J} = \phi_{NK-1,J} \quad \text{and} \quad \phi_{NK,J} = \phi_{2,J} \tag{9.31}$$

For the velocity component across the boundary we have

$$w_{1,J} = w_{NK-1,J} \quad \text{and} \quad w_{NK+1,J} = w_{3,J} \tag{9.32}$$

9.8 Potential pitfalls and final remarks

Flows inside a CFD solution domain are driven by the boundary conditions. In a sense the process of solving a field problem (e.g. a fluid flow) is nothing more than the extrapolation of a set of data defined on a boundary contour or surface into the domain interior. It is, therefore, of paramount importance that we supply physically realistic, well-posed boundary conditions, otherwise severe difficulties are encountered in obtaining solutions. The single, most common cause of rapid divergence of CFD simulations is the inappropriate selection of boundary conditions.

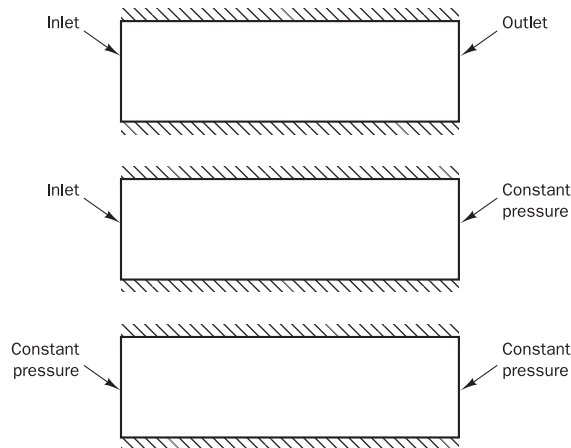
In Chapter 2 we summarised a set of ‘best’ boundary conditions for viscous fluid flows, which included the inlet, outlet and wall condition. Their

finite volume method implementation was discussed in sections 9.2 to 9.4, and in sections 9.5 to 9.7 we developed three further conditions, constant pressure, symmetry and periodicity, which are physically realistic and very useful in practical calculations. These are by no means the only boundary conditions. Commercial CFD packages may include time-dependent movement of boundaries, facilities to include rotating and accelerating boundaries and special conditions for transonic and supersonic flows. It would be beyond the scope of this book to discuss the ways of implementing all of them.

A simple illustration of poor selection of boundary conditions might be an attempt to generate a steady state solution in a domain with wall boundaries and a flow inlet but without an outlet boundary. It is obvious that mass cannot be conserved in the steady state and CFD calculations will 'blow up' swiftly. This almost trivial example also suggests that certain types of boundary conditions must be accompanied by particular other ones. We now briefly state some permissible combinations in subsonic flows:

- walls only
- walls and inlet and at least one outlet
- walls and inlet and at least one constant pressure boundary
- walls and constant pressure boundaries

Figure 9.17 illustrates these configurations for a simple duct flow.

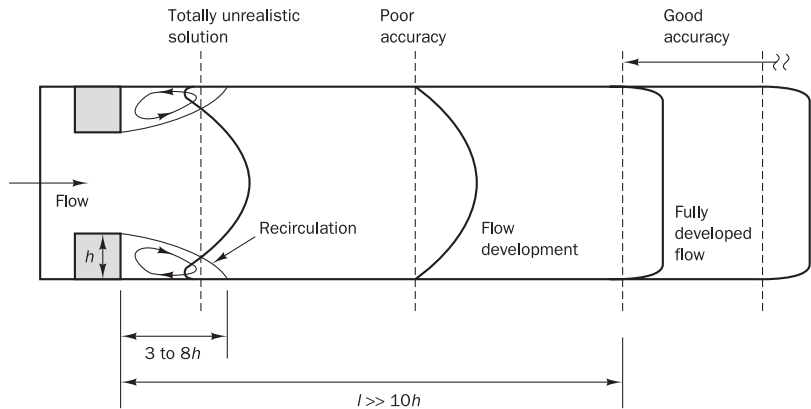
Figure 9.17 Configurations for a simple duct flow

Particular care must be taken in applying the outlet boundary condition. It can only be used if all flows entering the calculation domain are given by means of inlet boundary conditions (i.e. velocity and scalars fixed at inlet) and is only recommended for flow domains with a single exit. Physically the exit pressures govern the flow split between multiple outlets so it is better to specify this quantity at exits than (zero-gradient) outlet conditions. It is *not permitted* to combine an outlet condition with one or more constant pressure boundaries, because the zero-gradient outlet condition specifies neither the flow rate nor the pressure at the exit, thus leaving the problem under-specified.

We have glossed over a number of very complex problems by only considering subsonic flows. We merely warn the CFD user to tread very

carefully when attempting to tackle flows that may have regions of transonic and supersonic flows.

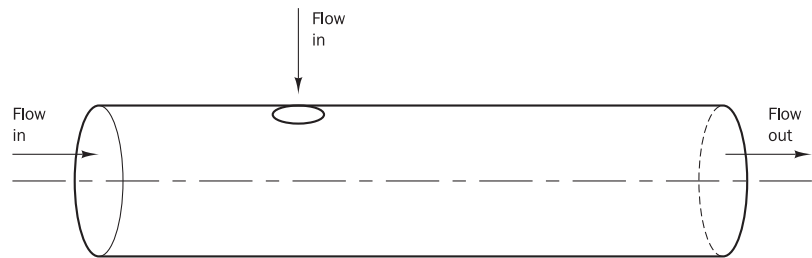
Accuracy limitations of the individual boundary conditions have already been pointed out before. Here we note a small selection of the more subtle pitfalls of practical CFD that need to be avoided to ensure that simulation accuracy is optimal:

- *Positioning of outlet boundaries.* If outlet boundaries are placed too close to solid obstacles it is possible that the flow has not yet reached a fully developed state (zero gradients in the flow direction), which may lead to sizeable errors. Figure 9.18 gives typical velocity profiles downstream of an obstacle, which illustrate the potential hazards.

Figure 9.18 Velocity profiles at different locations downstream of an obstacle

  If the outlet is placed close to an obstacle it may range across a wake region with recirculation. Not only does the assumed gradient condition not hold, but there is an area of reverse flow where the fluid enters the domain whilst we had assumed an outward flow. Of course, we cannot trust the solution if this condition arises. Somewhat further downstream there may not be reverse flow, but the zero-gradient condition does not hold since the velocity profile still changes in the flow direction. It is imperative that the outlet boundary is placed much further downstream than 10 heights downstream of the last obstacle to give accurate results. For high accuracy it is necessary to demonstrate that the interior solution is unaffected by the choice of location of the outlet by means of a sensitivity study for the effect of different downstream distances.
- *Near-wall grid.* The most accurate way of solving turbulent flows in a general-purpose CFD code is to make use of the good empirical fits provided by the wall function approach. To obtain the same accuracy by means of a simulation which includes points inside the (laminar) linear sub-layer the grid spacing must be so fine as to be uneconomical. The criterion that y^+ must be greater than 11.63 sets a *lower* limit to the distance from the wall Δy_P of the nearest grid point. The main mechanism for accuracy improvement available to us is grid refinement, but in a turbulent flow simulation we must ensure that, whilst refining the grid, the value of y^+ stays greater than 11.63 and is preferably between 30 and 500.

It is very often impossible to ensure that this is the case everywhere in a general flow; one pertinent example is a flow with recirculation. Near the reattachment point the velocity component parallel to the wall is zero, so by virtue of the criterion that y^+ must be greater than 11.63 the simulation reverts to the laminar case. There are additional problems associated with the k–ε model in these regions that give rise to further, even more important inaccuracies. Nevertheless, the point that it is difficult to keep y^+ above its lower limit is well illustrated.

- *Misapplication of the symmetry condition*. It is important to realise that geometric symmetry of the flow domain does not always imply that the flow possesses the same symmetry. An example shown in Figure 9.19 is the flow through a circular pipe with a side jet.

Figure 9.19 A non-symmetric flow situation in a cylindrical geometry

In spite of the fact that the domain has axisymmetry the occurrence of the cross-flow jet makes the flow non-axisymmetric. Although it is tempting to solve the problem in cylindrical polar co-ordinates, the flow solution will be inaccurate because flow may not cross the centreline.

We have discussed the implementation of the most important boundary conditions. Moreover, we have outlined suitable combinations of boundary conditions and highlighted particular problem areas. It is of crucial importance that the CFD user has a good understanding of all the relevant issues as a first step towards accurate flow simulations with the finite volume method.

Chapter ten Errors and uncertainty in CFD modelling

In this chapter we review:

- Why it is important to know about the error and uncertainty in CFD calculations
- Definitions and causes of error and uncertainty
- Methods to quantify error and uncertainty in CFD results: verification and validation
- Best practice in CFD as a systematic approach, which seeks to achieve the highest possible level of confidence in CFD simulation results for the available resources

10.1 Errors and uncertainty in CFD

During the 1990s the benefits of CFD were recognised by large corporations, small and medium-sized enterprises alike, and it is now used in design/development environments across a wide range of industries. This has focused attention on 'value for money' and the potential consequences of wrong decisions made on the basis of CFD results. The consequences of inaccurate CFD results are at best wasted time, money and effort and at worst catastrophic failure of components, structures or machines. Moreover, the costs of a CFD capability may be quite substantial:

- Capital cost of computing equipment
- Direct operating cost: software licence(s) and salary of CFD specialist(s)
- Indirect operating costs: maintenance of computing equipment and provision of information resources to support CFD activity

The value of a modelling result is clear – time savings in design and product improvement through enhanced understanding of the engineering problem under consideration – but rather harder to quantify. The application of CFD modelling as an engineering tool can only be justified on the basis of its accuracy and the level of confidence in its results. With its roots in academic research, CFD development was initially focused on new functionality and improved understanding without the need to make very precise statements relating to confidence levels. Engineering industry, however, has a long tradition of making things work within the limitations of the current state of knowledge, provided that the confidence limits are known. Assessment of uncertainty in experimental data, for example, is a well-established practice, and the relevant techniques (should) form part of every engineer's basic education.

To address the issue of trust and confidence in CFD the fraternity has now carried out extensive reviews of the factors influencing simulation results and developed a systematic process, akin to the estimation of uncertainty in

experimental results, for the quantitative assessment of confidence levels. This has led to the formulation of a number of guidelines for best practice in CFD, the most influential of which are the AIAA (1998) and ERCOFTAC (2000) guidelines. In this section we give a review of the most important concepts in the study of errors and uncertainty in CFD and summarise the recommendations for the conduct of CFD simulations contained in the two guides.

In the context of trust and confidence in CFD modelling, the following definitions of error and uncertainty have now been widely accepted (AIAA, 1998; Oberkampf and Trucano, 2002):

- ***Error***: a recognisable deficiency in a CFD model that is *not caused by lack of knowledge*. Causes of errors, defined in this way, are:
  (i) Numerical errors – roundoff errors, iterative convergence errors, discretisation errors
  (ii) Coding errors – mistakes or 'bugs' in the software
  (iii) User errors – human errors through incorrect use of the software
- ***Uncertainty***: a potential deficiency in a CFD model that is *caused by lack of knowledge*. The main sources of uncertainty are:
  (i) Input uncertainty – inaccuracies due to limited information or approximate representation of geometry, boundary conditions, material properties etc.
  (ii) Physical model uncertainty – discrepancies between real flows and CFD due to inadequate representation of physical or chemical processes (e.g. turbulence, combustion) or due to simplifying assumptions in the modelling process (e.g. incompressible flow, steady flow)

Coding and user errors are the most insidious forms of errors. The well-publicised failure on 23 September 1999 of NASA's Mars Climate Orbiter space mission was subsequently attributed to incompatibility between pieces of software written in SI and Imperial units, which shows that coding errors can catch out even the most sophisticated users and organisations. User error may be reduced or eliminated to a large extent through adequate training and experience. Systematic reduction of coding and user errors falls within the remit of software engineering/quality assurance. For the purposes of this introduction we assume that the code is correct and that user error is negligible. We focus our attention on the remaining unavoidable causes of errors and uncertainty and highlight their effects on CFD results. We describe the procedures for verification and validation of CFD aimed at quantitative assessment of errors and uncertainty in its results. Finally, we give a summary of available guidelines for best practice and make recommendations for the reporting of CFD model results.

10.2 Numerical errors

CFD solves systems of non-linear partial differential equations in discretised form on meshes of finite time steps and finite control volumes that cover the region of interest and its boundaries. This gives rise to three recognised sources of numerical error:

- Roundoff error
- Iterative convergence error
- Discretisation error

We briefly discuss each of these causes of error in turn and highlight methods to control their magnitude.

Roundoff errors

Roundoff errors are the result of the computational representation of real numbers by means of a finite number of significant digits, which is termed the machine accuracy. Roundoff errors contribute to the numerical error in a CFD result. These can generally be controlled by careful arrangement of floating-point arithmetic operations to avoid subtraction of almost equal-sized large numbers or addition of numbers with very large difference in magnitude. In CFD computations it is common practice to use gauge pressures relative to a specified base pressure (e.g. in incompressible flow simulations a zero pressure value is set at an arbitrary location within the computational domain). This is a simple example of error control by good code design, since it ensures that the pressure values within the domain are always of the same order as the pressure difference that drives the flow. Thus, the calculation with floating-point arithmetic of pressure differences between adjacent mesh cells is not spoilt by loss of significant digits as would be the case if they were evaluated as the difference between comparatively large absolute pressures.

Iterative convergence errors

Figures 6.6–6.8 show that the numerical solution of a flow problem requires an iterative process. The final solution exactly satisfies the discretised flow equations in the interior of the domain and the specified conditions on its boundaries. If the iteration sequence is convergent the difference between the final solution of the coupled set of discretised flow equations and the current solution after k iterations reduces as the number of iterations increases. In practice, the available resources of computing power and time dictate that we truncate the iteration sequence when the solution is sufficiently close to the final solution. This truncation generates a contribution to the numerical error in the CFD solution.

Before moving on we briefly consider methods used in CFD codes to truncate the iterative process. To determine whether it is worth making additional effort to get closer to the final solution we would ideally like a truncation criterion in the form of a single number that can be tested against a pre-set tolerance. There are several different ways of constructing practically useful truncation criteria in CFD, but by far the most common one is based on so-called residuals. The discretised equation for general flow variable ϕ at mesh cell i can be written as follows:

$$(a_P\phi_P)_i = \left(\sum_{nb} a_{nb}\phi_{nb}\right)_i + b_i \tag{10.1}$$

where subscript i indicates the control volume

The final solution will satisfy equation (10.1) exactly at all cells in the mesh, but after k iterations there will be a difference between the left and right hand sides. The **absolute value** of this difference at mesh cell i is termed the **local residual** R_i^ϕ:

$$(R_i^{\phi})^{(k)} = \left| \left(\sum_{nb} a_{nb}\phi_{nb} \right)_i^{(k)} + b_i^{(k)} - (a_P\phi_P)_i^{(k)} \right| \tag{10.2}$$

where superscript (k) indicates the current iteration count

To get an indication of the convergence behaviour across the whole flow field, we define the global residual $\hat{R}^{\phi}$, which is just the sum of the local residuals over all M control volumes within the computational domain. After k iterations we have

$$(\hat{R}^{\phi})^{(k)} = \sum_{i=1}^{M} (R_i^{\phi})^{(k)} = \sum_{i=1}^{M} \left| \left(\sum_{nb} a_{nb}\phi_{nb} \right)_i^{(k)} + b_i^{(k)} - (a_P\phi_P)_i^{(k)} \right| \tag{10.3}$$

Note that the absolute value in the definition of the local residual prevents cancellation of positive and negative contributions of similar size, which would result in a zero global residual whilst some or all of the local residuals are non-zero.

Inspection of equation (10.3) shows that the magnitude of the global residual $\hat{R}^{\phi}$ decreases as we get closer to the final solution, since the size of the local residuals should decrease in a converging sequence. Thus, it would seem that $\hat{R}^{\phi}$ might be a satisfactory single number indicator of convergence. However, the global residual will be larger in simulations where the flow variable ϕ has a larger magnitude, so we would need to specify different truncation values for $\hat{R}^{\phi}$. This can be resolved if we use a global residual that is scaled to take out the magnitude of ϕ. Thus, we define the normalised global residual $\hat{R}_N^{\phi}$ for flow variable ϕ after k iterations as follows:

$$(\hat{R}_N^{\phi})^{(k)} = (\hat{R}^{\phi})^{(k)} / \hat{F}_{R\phi} \tag{10.4}$$

where $\hat{F}_{R\phi}$ is the normalisation factor

The normalisation factor $\hat{F}_{R\phi}$ is a reference level of the residuals for flow variable ϕ. Three common normalisation methods are given below:

$$\hat{F}_{R\phi} = (\hat{R}^{\phi})^{(k_0)} \qquad \Rightarrow (\hat{R}_N^{\phi})^{(k)} = (\hat{R}^{\phi})^{(k)} / (\hat{R}^{\phi})^{(k_0)} \tag{10.5a}$$

$$\hat{F}_{R\phi} = \sum_{j}^{\text{inlet cells}} (\rho A \mathbf{U}\,.\,\mathbf{n})_j \phi_j \Rightarrow (\hat{R}_N^{\phi})^{(k)} = (\hat{R}^{\phi})^{(k)} \Big/ \sum_{j}^{\text{inlet cells}} (\rho A \mathbf{U}\,.\,\mathbf{n})_j \phi_j \tag{10.5b}$$

$$\hat{F}_{R\phi} = \sum_{i=1}^{M} \left| (a_P\phi_P)_i^{(k)} \right| \qquad \Rightarrow (\hat{R}_S^{\phi})^{(k)} = (\hat{R}^{\phi})^{(k)} \Big/ \sum_{i=1}^{M} \left| (a_P\phi_P)_i^{(k)} \right| \tag{10.5c}$$

In definition (10.5a) the global residual is normalised by its own size at iteration k_0 ($k_0 \neq 1$ and usually < 10). In (10.5b) the total rate of flow of ϕ into the domain is used as the normalising factor. Finally, definition (10.5c) uses the absolute value of the left hand side of equation (10.1) summed over all mesh cells. The three different choices of normalisation factor each have advantages and disadvantages in specific cases. Whichever definition is used, the normalised global residual is always equal to zero when the final solution is reached. Moreover, $\hat{R}_N^{\phi}$ does not require case-by-case adjustment, so it is a satisfactory average measure of the discrepancy between the final solution and the computed solution after k iterations.

In commercial CFD codes the convergence test in the iterative sequences (see Figures 6.6–6.8) involves specification of tolerances for the normalised global residuals for mass, momentum and energy. An iteration sequence is automatically truncated when all these residuals are smaller than their pre-set maximum value. Default values for the tolerances, which have been determined by systematic trials to give acceptable results for a wide range of flows, are supplied by the code vendors. For high-accuracy work it may be necessary to reduce the values of these tolerances from their default values to control and reduce the magnitude of the contribution to the numerical error due to early truncation of the iterative sequence.

Discretisation errors

Temporal and spatial derivates of the flow variable, which appear in the expressions for the rates of change, fluxes, sources and sinks in the governing equations, are approximated in the finite volume method on the chosen time and space mesh. We have shown in Chapters 4 and 5 that this involves simplified profile assumptions for flow variable ϕ, and Appendix A shows that this practice corresponds to the truncation of a Taylor series. The discretisation error is associated with the neglected contributions due to the higher-order terms, which gives rise to errors in CFD results. Control of the magnitude and distribution of discretisation errors through careful mesh design is a major concern in high-quality CFD. In theory, we can make the discretisation error arbitrarily small by progressive reductions of the time step and space mesh size, but this requires increasing amounts of memory and computing time. Thus, the ingenuity of the CFD user as well as resource constraints dictate the lowest achievable level of the contribution to the numerical error due to the simplified profile assumptions.

10.3 Input uncertainty

Input uncertainty is associated with discrepancies between the real flow and the problem definition within a CFD model. We consider data inputs under the following headings:

- Domain geometry
- Boundary conditions
- Fluid properties

Below we give examples of the factors that can lead to uncertainty in CFD results for each of these three categories of input data.

Domain geometry

The definition of the domain geometry involves specification of the shape and size of the region of interest. In industrial applications this may come from a CAD model of, say, a flow duct. It is impossible to manufacture the duct perfectly to the design specifications; manufacturing tolerances will lead to discrepancies between the design intent and a manufactured part. Furthermore, the CAD model needs to be converted to be suitable within CFD. This conversion process can lead to discrepancies between the design intent and the geometry within CFD. Similar comments apply to the surface roughness. Finally, the boundary shape in CFD is a discrete representation

of the real boundary, e.g. by means of straight lines or simple curves connecting boundary nodes. In summary, the macroscopic and microscopic geometry within the CFD model will be somewhat different from the real flow passage, which contributes to input uncertainty in the model results.

Boundary conditions

Apart from the shape and surface state of solid boundaries, it is also necessary to specify the conditions on the surface for all other flow variables, such as velocity, temperature, species etc. It can be difficult to acquire this type of input to a high degree of accuracy. Simple assumptions, e.g. given temperature, given heat flux, adiabatic wall, are often made in the computations; the accuracy of these will affect the calculation result.

The choice of type and location of open boundaries through which flow enters and leaves the domain is a particular challenge in CFD modelling. Boundary conditions are chosen from a limited set of available boundary types. In Chapter 2 we reviewed the main conditions at flow inlets: (a) fixed pressure, (b) fixed mass flow rate, or (c) given distributions of velocity and turbulence parameters. At flow outlets we can specify (i) pressure in conjunction with any of the inlet conditions or (ii) an outflow boundary condition (zero rate of change in the flow direction for all flow variables) in conjunction with specified mass flow rate (b) or velocity (c).

There must be compatibility between the chosen open boundary condition type and the flow information available on the chosen surface location. In some cases, we only have partial information, e.g. average velocity and some indication of velocity distribution but no information on the turbulence parameters. Missing information must now be generated on the basis of past experience or inspired guesswork. In other cases, the assumed boundary condition may only be approximately true. For example, the pressure is assumed to be uniform on a fixed pressure boundary, but might actually be somewhat non-uniform. A contribution to the input uncertainty is associated with the inaccuracy of all assumptions involved in the process of defining the boundary conditions.

The location of the open boundaries must be sufficiently far from the area of interest so that it does not affect the flow in this region. Solution economy on the other hand dictates that the domain should not be excessively large, so a compromise must be found, which may cause discrepancies between the real flow and the CFD model, resulting in a contribution to the input uncertainty.

Fluid properties

All fluid properties (e.g. density, viscosity, thermal conductivity) depend to a greater or lesser extent on the local value of flow parameters, such as pressure and temperature. Often the assumption of a constant fluid property is acceptable provided that the spatial and temporal variations of the flow parameters influencing that property are small. The application of this assumption also benefits solution economy, since CFD models converge more quickly if fluid properties remain constant; however, errors are introduced if the assumption of constant fluid properties is inaccurate. If the fluid properties are allowed to vary as functions of flow parameters we have to

contend with errors due to experimental uncertainty in the relationships describing the fluid properties.

10.4 Physical model uncertainty

Limited accuracy or lack of validity of submodels

CFD modelling of complex flow phenomena, such as turbulence, combustion, heat and mass transfer, involves semi-empirical submodels. They encapsulate the best scientific understanding of complex physical and chemical processes. The submodels invariably contain adjustable constants derived from high-quality measurements on a limited class of simple flows. In applying the submodels to more complex flows we extrapolate beyond the range of these data. Doing this, it is tacitly assumed that the physics/chemistry does not change too much, so that (i) the submodel still applies and (ii) the values of adjustable constants do not need to change. There are several reasons why the application of submodels brings uncertainty in a CFD result:

- A complex flow may involve entirely new and unexpected physical/chemical processes that are not accounted for in the original submodel. In the absence of a better submodel, the user has no option but to work with a less sophisticated description of the flow.
- In spite of the availability of a more comprehensive submodel, the user may deliberately select a simpler submodel with a less accurate account of physics/chemistry, e.g. to save time in computations.
- A complex flow may include the same mixture of physics/chemistry as the original simple flows, but not exactly in the same blend, requiring adjustments of the submodel constants.
- The empirical constants within the submodels represent a best fit of experimental data, which will themselves have some uncertainty.

To clarify some of these points, we discuss the causes of physical model uncertainty in the k–ε turbulence model, which was introduced in Chapter 3. It is a two-equation turbulence model with five adjustable constants: C_μ, σ_k, σ_ε, $C_{1\varepsilon}$, $C_{2\varepsilon}$. This model is semi-empirical, since the values of these five constants have been calibrated to match results for decay of isotropic turbulence and properties of thin shear layers such as boundary layers where turbulence production and dissipation are nearly in balance. The k–ε model is used as an industry standard since it is comparatively cheap to run and gives acceptable results in many cases. Its performance has been assessed extensively and its flaws are well documented. It performs well for flows that are fairly close to the cases used to calibrate the model constants, but is less accurate when tackling flows with more complex strain fields, e.g. boundary layers with large adverse pressure gradients, separated and reattaching flows, strongly swirling flows etc. In such flows some of the physical processes that affect turbulence parameters and, hence, the entire flow field are not captured within the k–ε modelling framework. This leads to a contribution to physical modelling uncertainty.

The standard k–ε model includes the wall function approach. This is a computationally economical method, which avoids having to resolve the entire boundary layer profile by representing the properties of near-wall turbulent boundary layers by means of algebraic relationships. The log-law is itself an empirical description of flow behaviour. Moreover, the constant E

in log-law equation (3.49) must be adjusted to account for the roughness of the wall surface. As noted in section 3.7.2, there are stringent requirements on the placement of near-wall grid points, which should be located at a non-dimensional distance from the wall within the range $30 < y^+ < 500$. In a complex 2D or 3D flow with separation and reattachment it is impossible to satisfy the y^+ requirements everywhere, and there will be local violations that may also affect downstream flow development, giving rise to further contributions to the physical model uncertainty.

Finally, we have already noted that the log-law only describes turbulent boundary layers with modest pressure gradients at high Reynolds numbers. Additional techniques have since been developed to cope with low Reynolds number turbulence and flows where it is deemed necessary to resolve the entire boundary layer profile. In Chapter 3 we discussed low Reynolds number k–ε models. Currently, the most popular method is to use the two-layer model whereby the properties of the near-wall region are not evaluated by means of algebraic relations, but extracted from the solution of a one-equation turbulence model. In this case the near-wall grid points must be positioned such that $y^+ < 1$ and at least 10–20 points are employed to resolve the boundary layer profile. Careful attention must be paid to meshing detail to avoid violation of these requirements.

Other turbulence modelling options within commercial CFD codes include one-equation models (e.g. the Spalart–Allmaras model), other two-equation models (e.g. the k–ω model), the Reynolds stress model (RSM) and large eddy simulation (LES). They all contain adjustable constants and, hence, they can only capture exactly the class of flows that were used to calibrate their values. Besides turbulence models, commercial CFD codes also contain a range of submodels for other important applications areas, e.g. combustion. Each submodel will contain empirical constants that have limited validity. In summary, the empirical nature of the submodels inside a CFD code, the experimental uncertainty of the values of the submodel constants and the appropriateness of the chosen submodel for the flow to be studied together determine the level of errors in the CFD results due to physical model uncertainty.

Limited accuracy or lack of validity of simplifying assumptions

At the start of each CFD modelling exercise it is common practice to establish whether it is possible to apply one or more potential simplifications. Considerable solution economy can be achieved if the flow can be treated as:

- Steady vs. transient
- Two-dimensional, axisymmetric, symmetrical across one or more planes vs. fully three-dimensional
- Incompressible vs. compressible
- Adiabatic vs. heat transfer across the boundaries
- Single species/phase vs. multi-component/phase

In many cases it is relatively easy to see if a simplification is justifiable to good accuracy. For example, the validity of the incompressible flow assumption depends on the value of the Mach number M. The differences between incompressible and compressible CFD simulations are slight when $M < 0.3$. As M gets closer to unity the discrepancy between the two approaches gradually becomes larger, and hence the physical model uncertainty associated

with the incompressible assumption will increase. Near $M = 1$ a CFD result based on the incompressible flow assumption becomes meaningless, since shocks cannot be reproduced.

In other cases, matters are less straightforward. Many flows exhibit geometrical symmetry about one or two planes. However, unless the inlet flow possesses the same symmetry, a model simplification based on geometrical symmetry will be inaccurate. Some flows through symmetrical passages are sensitively dependent on inflow conditions, e.g. the flow through gradual area enlargements (diffusers). It is tempting to simplify a CFD model by approximating an almost uniform inflow into a symmetrical domain by means of a uniform one. However, if the divergence angle of a planar diffuser is within the range 20°–60° the small asymmetry in the incoming flow will be amplified and cause the flow to attach to one of the side walls accompanied by reverse flow on the opposite wall. This will, of course, lead to major discrepancies between the real flow and a CFD result based on the symmetry assumption.

A different type of problem is encountered when we consider the steady, uniform oncoming flow around a cylinder with axis perpendicular to the flow. For a very wide range of velocities, a periodic wake flow develops behind the cylinder, known as the von Karman vortex street. Simulations with steady flow and/or symmetry assumption would fail to capture this phenomenon, with attendant loss of simulation accuracy.

The accuracy and appropriateness of all simplifying assumptions for a given flow determine the size of their contribution to physical model uncertainty.

10.5 Verification and validation

Once it is recognised that errors and uncertainty are unavoidable aspects of CFD modelling, it becomes necessary to develop rigorous methods to quantify the level of confidence in its results. In this context, the following terminology due to AIAA (1998) and Oberkampf and Trucano (2002) has now been widely accepted:

- ***Verification***: the process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution to the model. Roache (1998) coined the phrase 'solving the equations right'. This process quantifies the errors.
- ***Validation***: the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. Roache (1998) called this 'solving the right equations'. This process quantifies the uncertainty.

Below we discuss the methods of verification and validation.

Verification

The process of verification involves quantification of the errors. Since we are ignoring computer coding errors and user errors, we need to estimate the roundoff error, iterative convergence error and discretisation error.

- *Roundoff error* can be assessed by comparing CFD results obtained using different levels of machine accuracy (e.g. in single precision,

7 significant figures in Fortran; or double precision, 16 significant figures in Fortran).

- *Iterative convergence error* can be quantified by investigating the effects of systematic variation of the truncation criteria for all residuals on target quantities of interest, e.g. the computed pressure drop or mass flow rate in an internal flow, the force on an object in an external flow, the velocity at one or more locations of interest. Differences between the values of a target quantity at various levels of the truncation criteria provide a quantitative measure of the closeness to a fully converged solution.
- *Discretisation error* is quantified by systematic refinement of the space and time meshes. In high-quality CFD work we should aim to demonstrate monotonic reduction of the discretisation error for target quantities of interest and the flow field as a whole on **two or three successive levels of mesh refinement**. We briefly describe the methods used for discretisation error estimation.

We assume that the numerical solution satisfies the following conditions (Roache, 1997):

- The flow field is sufficiently smooth to justify the use of Taylor series expansions (i.e. no discontinuities in any of the flow variables)
- The convergence is monotonic (i.e. if the value of a target quantity increases/reduces by an amount X upon going from a coarse mesh to a medium mesh, its value should again increase/reduce upon going from the medium mesh to a fine mesh and the magnitude of the change should be smaller than the magnitude of X)
- The numerical method is in its asymptotic range (i.e. the leading term of the Taylor series expansion dominates the truncation error behaviour)

If we consider the numerical solution of a steady flow problem under the stated conditions we can write the following estimate of the error E_U in a target quantity U as a function of a reference size h of the control volumes inside the mesh:

$$E_U(h) = U_{exact} - U \approx Ch^p \tag{10.6}$$

where C is a constant and p is the order of the numerical scheme

For two meshes with refinement ratio $r = h_2/h_1$ and solutions U_1 and U_2 it is easy to show that the estimate of the discretisation error can be written in terms of the difference $U_2 - U_1$ between the two solutions:

$$E_{U,1} = \frac{U_2 - U_1}{1 - r^p} \tag{10.7a}$$

$$E_{U,2} = r^p\left(\frac{U_2 - U_1}{1 - r^p}\right) \tag{10.7b}$$

where $E_{U,1}$ is the error in the coarse solution and
$E_{U,2}$ is the error in the fine mesh solution

Similar grid refinement techniques can be used to estimate the discretisation error due to the finite time step size. Roache (1997) also gave an error estimate for fully implicit transient solutions based on an additional explicit solution

using the original time step size, which is much more economical than time step refinement.

Roache (1997) noted that the estimates of equations (10.7a–b) are approximate and do not constitute bounds on the discretisation error. He proposed a so-called grid convergence indicator (GCI) to quantify the numerical error in a CFD solution:

$$\mathrm{GCI}_U = F_S E_U \tag{10.8}$$

where F_S is the safety factor

A conservative value of safety factor $F_S = 3$ is suggested.

Roache also noted that it should not be taken for granted that the actual truncation error in a numerical solution will decay exactly in accordance with the formal order p of accuracy of the basic numerical scheme. He gave several examples where his investigations had shown this not to be the case due to seemingly minor flaws in the numerical method, and advocated using the observed order of truncation error decay on three successively refined meshes. For constant refinement ratio $r = h_2/h_1 = h_3/h_2$ the observed order $\tilde{p}$ of the truncation rate decay can be found as follows:

$$\tilde{p} = \ln\left(\frac{U_3 - U_2}{U_2 - U_1}\right) \Big/ \ln(r) \tag{10.9}$$

where $U_2 - U_1$ is the difference between the solutions on the medium and coarse mesh

and $U_3 - U_2$ is the difference between the solutions on the fine and medium mesh

In codes with flaws the observed value of truncation error reduction rate $\tilde{p}$ is always smaller than the formal order of accuracy p of the underlying numerical schemes. In high-quality studies using two or more levels of refinement, it is recommended that discretisation error formulae (10.7a–b) should be evaluated using the observed value $\tilde{p}$ from equation (10.9) and used in conjunction with a reduced safety factor $F_S = 1.25$ in grid convergence index formula (10.8).

Finally, we note that the above methods merely estimate the numerical error of the code as it is and do not test whether the code itself accurately reflects the mathematical model of the flow envisaged by the code designer. Oberkampf and Trucano (2002), therefore, argued that a complete programme of verification activities should always include a stage of systematic comparison of CFD results with reliable benchmarks, i.e. highly accurate solutions of (usually simple) flow problems, such as analytical solutions or highly resolved numerical solutions.

Validation

The process of validation involves quantification of the input uncertainty and physical model uncertainty.

- *Input uncertainty* can be estimated by means of sensitivity analysis or uncertainty analysis. This involves multiple test runs of the CFD model with different values of input data sampled from probability distributions based on their mean value and expected variations.

The observed variations of target quantities of interest can be used to produce upper and lower bounds for their expected range and, hence, are a useful measure of the input uncertainty. In sensitivity analysis the effects of variations in each item of input data is studied individually. Uncertainty analysis, on the other hand, considers possible interactions due to simultaneous variations of different pieces of input data and uses Monte Carlo techniques in the design of the programme of CFD test runs.

- Oberkampf and Trucano (2002) stated that quantitative assessment of the *physical modelling uncertainty* requires comparison of CFD results with high-quality experimental results. They also noted that meaningful validation is only possible in the presence of good quantitative estimates of (i) all numerical errors, (ii) input uncertainty and (iii) uncertainty of the experimental data used in the comparison.

Thus, the ultimate test of a CFD model is a comparison between its output and experimental data. However, the way in which such a comparison should best be carried out is still a subject of discussion. The most common way of reporting the outcome of a validation exercise is to draw a graph of a target quantity (say, the discharge coefficient of an orifice, the force on an object in the flow) on the y-axis and a flow parameter (say, flow velocity or Reynolds number) on the x-axis. If the difference between computed and experimental values looks sufficiently small the CFD model is considered to be validated. The latter judgement is rather subjective, and Coleman and Stern (1997) proposed a more rigorous basis for validation comparisons drawing on the practice of estimating uncertainty in experimental results involving several independent sources of uncertainty. They suggested that the errors should be combined statistically by calculating the sum of squares of estimates of numerical errors, input uncertainty and experimental uncertainty to form an estimate of validation uncertainty. A simulation is considered to be validated if the difference between experimental data and CFD model results is smaller than the validation uncertainty. The level of confidence in the CFD model is indicated by the magnitude of the validation uncertainty.

Oberkampf and Trucano (2002) pointed out that this approach would have the slightly paradoxical implication that it is easier to validate a CFD result with poor-quality experimental results containing a large amount of scatter. They suggested an alternative validation metric, which includes a statistical contribution, the influence of which decreases as the variance of the experimental data decreases with increase of the number of repeat experiments. Thus, the metric indicates increased levels of confidence in a validated CFD code if (i) the difference between the experimental data and CFD results is small and (ii) the experimental uncertainty is small.

Irrespective of their individual merits, both methods provide a more objective basis for validation comparisons, but interested readers are directed to watch out for further developments since this topic is still in its early stages.

Data sources for verification and validation

By now it should be clear that the accuracy of a CFD result cannot be taken for granted, and verification and validation are mission-critical elements of the confidence-building process. For this we require experimental data with

(i) comprehensive documentation of problem geometry and boundary conditions, (ii) detailed measurements of distributions of flow properties, such as velocity components, static/total pressure, temperature etc., and (iii) complementary overall measurements, e.g. mass flow rate, overall pressure drop etc. Naturally, we should limit ourselves to information from trusted sources to generate a sufficiently credible validation. Several public-access databanks have now emerged to support CFD validation work. The most prominent ones are:

- ERCOFTAC: http://ercoftac.mech.surrey.ac.uk/ – the leading database with links to refereed experimental datasets and high-quality CFD simulations including LES and DNS
- NASA: http://www.larc.nasa.gov/reports/reports.htm/ – NACA and NASA reports in downloadable form
- Flownet: http://dataserv.inria.fr/flownet/ – EU database
- Overview of current validation resources: http://www.cfd-online.com/Resources/refs.html

The following journals have been useful to authors in the past:

- *Annual Review of Fluid Mechanics*
- *Journal of Fluid Mechanics*
- *AIAA Journal*
- *Journal of Fluids Engineering, Transactions of the ASME*
- *Journal of Heat Transfer, Transactions of the ASME*
- *International Journal of Heat and Mass Transfer*
- *International Journal of Heat and Fluid Flow*
- *Combustion and Flame*
- *Physics of Fluids*
- *Experiments in Fluids*
- *International Journal of Wind Engineering and Industrial Aerodynamics*
- *Journal of Power Engineering, Transactions of the ASME*
- *Journal of Turbomachinery, Transactions of the ASME*
- *Proceedings of the IMechE, Part C: Journal of Mechanical Engineering Science*

If suitable experimental results for a comprehensive validation are not available, it will be necessary to identify a dataset for a closely related problem. If the problem chosen for validation is sufficiently close to the actual problem to be studied, we should be able to apply roughly the same CFD approach in both cases. However, we have already seen that some flow problems can be very sensitive to apparently minor changes in the boundary conditions or problem geometry. So care must be taken in the formulation of validation cases, and past experience should play an important role in the justification of the chosen approach.

In the absence of high-quality measurements, we may have to settle for comparison of CFD output against other data in academic or industrial journals. The Engineering Science Data Unit (ESDU) database also provides a particularly comprehensive collection of carefully refereed engineering design information with many fluid flow data items (http://www.esdu.com/ – readers should be aware that a subscription fee is payable for access to this database). Finally, it should be noted that a sufficient level of confidence in CFD simulations can only be achieved through rigorous verification and validation. If the search for validation data draws a complete blank it is

essential that a reasonable programme of experimentation be undertaken alongside CFD to provide solid foundations for design recommendations.

10.6 Guidelines for best practice in CFD

Our review of the subject has shown that the scope for errors and uncertainty in the study of complex industrial systems with CFD is huge, due to the large variety of user inputs and modelling choices that must be made. The purpose of verification and validation activities is the quantification of errors and uncertainty. Best practice guidelines, on the other hand, seek to define routes to maximise the accuracy and level of confidence in CFD models within the constraints of existing knowledge and available resources. We review the two most influential sets of guidelines, AIAA (1998) and ERCOFTAC (2000), which set out the main rules for the conduct of CFD modelling studies with a view to confidence building in industrial applications. In the previous material of this section we have broadly followed the development of the concepts of error, uncertainty, verification and validation in these references. Here we highlight some further aspects of the two sets of guidelines that have not already been covered in this section.

AIAA guide (1998)

The AIAA guide was the first to be compiled and gives particular emphasis to CFD in complex systems. It develops the foundations of quality assurance and software engineering in CFD through cross-references with sources in other fields and explains the approach to verification and validation for confidence building in CFD results. In addition to this, two further issues are raised that require careful attention:

- The AIAA guide notes that the processes of verification and validation can only demonstrate satisfactory performance of a CFD code **for specific instances** of its use through comparisons of the code output with high-quality benchmark solutions and high-quality experiments. This notion stems from the complexity of industrial fluid flow problems and the wide range of numerical parameters that need to be selected as user inputs to generate CFD results.

This cautious position implies that, given the present state of the art, it is not possible to validate a CFD code for a new real-life industrial flow problem without high-quality experimental data for this actual problem. In our summary of validation we have suggested that we would probably be confident of a satisfactory outcome of a CFD model of a new, real-life problem if we had demonstrated satisfactory performance in a case that is in some sense sufficiently close. We believe that this is justifiable if there is a substantial peripheral knowledge base available, e.g. in the form of background knowledge of the fundamentals of fluid dynamics and associated topics, experimental data and operating experience with existing designs or similar devices on the market. In an industrial setting this is often the case and can be exploited to help decide which problem is sufficiently well understood and close to a new flow problem to help with validation activities.

The AIAA guide also contains specific recommendations for the conduct of modelling studies in cases with complex systems for which it is recognised that it is impossible or too expensive to obtain high-quality data on the full system. Below we summarise the approach:

- The credibility of the predictions of a CFD model is directly affected by the level of complexity of the problem to be tackled. Real-life flow problems include many sources of complexity associated with multi-dimensional and/or unsteady flow, geometric complexity, complex flow physics and/or chemistry. It is unrealistic to expect the same level of confidence for CFD models of very complex systems (aeroengine, furnace etc.) and of simple unit problems (e.g. internal flow through a straight pipe or orifice, external flow around an aerofoil or obstacle). The AIAA guide suggests that a **building block approach** be applied to the modelling of complex systems. The complexity of the full system is reduced by decomposition into simpler sub-systems. This process of complexity reduction is carried through in successive stages and ends with the identification of a series of simple unit problems for which high-quality experimental data *are* available and, therefore, comprehensive validation is possible. Lessons learnt in connection with the conduct of CFD simulations (numerical parameter choices, meshing practice etc.) should be implemented as the study progresses back upwards through the various stages of sub-systems in the direction of increased complexity. At each stage CFD results are compared with experimental data to refine the modelling approach, whilst taking note that problem definition and measured data are likely to be less precise as we approach the real-life flow system.

Thus, the AIAA guide provides a comprehensive strategy for the modelling of complex industrial flow problems, which (i) builds on strong foundations of well-validated simple unit problems, (ii) systematically increases the complexity of the models, (iii) incorporates all learning experiences and (iv) exploits the maximum number of opportunities for validation on the way from simple unit problems to the full problem.

ERCOFTAC guidelines (2000)

The ERCOFTAC guidelines provide an authoritative set of best practice rules for the conduct of less complex flow problems. The focus is on the prediction of single phase fluid flow and heat transfer and methods to quantify and minimise all sources of error and uncertainty. The document contains an extended section on the application of classical turbulence models, i.e. those based on Reynolds-averaged Navier–Stokes equations. The guide is aimed at less experienced users, and its practical implementation in CFD modelling is facilitated by the provision of extensive checklists. Moreover, eight case studies are presented with application of the guidelines and demonstrations of the achievable accuracy in flows ranging in complexity from a sudden pipe expansion to a low-speed centrifugal compressor.

We would encourage all readers to strive to develop a high-quality CFD approach based on the AIAA and ERCOFTAC guidelines and urge them to consult both references and further industry-specific guidelines such as MARNET-CFD (https://pronet.wsatkins.co.uk/marnet/guidelines/guide.html), Chen and Srebric (2001, 2002) and Srebric and Chen (2002) for more detailed advice. Moreover, we also draw attention to emerging networks, such as QNET (http://www.qnet-cfd.net/) and eFluids (http://www.efluids.com/), devoted to the dissemination of information relating to fluid mechanics as well as best practice in CFD.

10.7 Reporting/documentation of CFD simulation inputs and results

In order to open up CFD simulations to independent scrutiny within industrial organisations it is essential to have a comprehensive and uniform system of reporting. This is also useful as a basis for archiving simulations for future use with the ultimate aim of preserving past learning experiences and spreading best practice throughout groups of users within an organisation. First we suggest a list of items necessary for documentation of user input.

Input documentation

- General description of the problem and purpose of CFD simulation
- Code chosen for solution of problem
- Computing platform used for run
- Schematic diagram of the region of interest with all key dimensions, flow inlets and outlets
- Boundary conditions – include comments on/justifications of assumptions made and known areas of approximation or lack of information
- Initial conditions for transient flow simulations or field initialisations for steady flow studies
- Fluid properties – include comments/justifications of assumptions and data sources
- Modelling option selections: (i) laminar/turbulent + turbulence model + near-wall treatment, (ii) combustion model, (iii) other physical models – with comments/justifications on selections
- Grid design: temporal mesh, space mesh including one or more diagrams of the grid that are sufficiently clear to illustrate the approach to mesh design – also give written comments on compromises and details of grid-independence study
- Solution algorithm choices – particularly non-default choices; note that default settings may change as a CFD code evolves, so a comprehensive summary of all the main selections (first/second-order schemes, multigrid options, segregated/coupled solver etc.) is preferable for long-term archiving
- Iterative convergence criteria choices: settings of truncation levels for residuals and choice of additional target quantities for convergence monitoring
- Brief summary of particular aspects of simulation design that required special attention to get simulation to work and to get accurate results, also noting unresolved problem issues

Next, we list items to assist scrutiny and confidence building in result analysis and reporting.

Result interpretation and reporting

Alongside options for alphanumeric output, commercial CFD codes have the ability to produce a wide variety of result visualisations, such as:

- Velocity vector plots
- Streaklines and particle paths
- Contour plots of flow variable
- Profile plots

- Grid display
- View manipulation

It is important to note that high-quality **presentation** is not necessarily synonymous with high-quality **results**. Less experienced users should not be taken in by the power of the post-processing capabilities of a CFD code. Before communicating the findings of a CFD study and drawing conclusions it is essential that the quality of the results is checked thoroughly by verification and validation. Below, we summarise the main elements to be checked and documented:

- *Verification study*: give estimates of numerical errors. For high-accuracy work all key target quantities should be shown to be independent of iterative convergence criterion and mesh. Diagrams of spatial distributions of residuals can help illustrate regions of unacceptably high residuals even if global residuals are sufficiently low to indicate iterative convergence. Identify where compromises were necessary if the results are still grid dependent.
- *Quantification of input uncertainty*: the main problem is generally specification of the boundary conditions; where necessary also consider fluid properties.
- *Validation study*: summarise method used to validate CFD approach; outline how the AIAA building block approach was applied if a complex system is studied; give comments on how improved match with experimental data was achieved by changes to the modelling strategy.
- *Further confidence* in the results can be built by analysis of the results using basic knowledge of fluid dynamics and conservation laws. This might involve consistency checks to identify where results are different from expectations. An obvious check would be a test of global mass, momentum, energy and species conservation by balancing the fluxes in and out of the region of interest with the sum of all sources and sinks inside the domain.

We have previously noted that time constraints and computer resources often determine the acceptable degree of convergence of a CFD simulation. This means that global conservation checks will not show exact balance of all the relevant fluxes and rates of creation and destruction. However, a significant departure from global conservation indicates problems.

Whilst we have made it clear that the only true quality check is validation, it is advisable to apply a range of common-sense quality tests where new flow problems are investigated. These can be based on a general understanding of fluid mechanics and/or specific knowledge of the application that is being studied. Here we give some items (trivial and profound) that might be checked when the outcome of a CFD simulation is evaluated:

- Fluid flows from high to low pressure (in pressure-driven flows)
- Static pressure decreases when velocity increases (Bernoulli's theorem for inviscid flows)
- Friction losses cause a decrease of total pressure in the direction of flow (viscous flow)
- Entropy must increase in the flow direction in a flow without heat transfer (consequence of second law of thermodynamics)
- The speed of a fluid near a stationary wall is smaller than the speed further away from the wall (boundary layer formation)

- Flow adopts a fully developed state after a sufficiently long distance in a straight duct with constant cross-section
- Boundary layers rapidly separate under the influence of an adverse pressure gradient (pressure increases in the direction of flow outside the boundary layer)
- Flows will usually separate at corners
- If a flow separates there is always recirculation
- A flow emerging into a large expanse of fluid from a small hole generally forms a jet
- Pressures are higher at the outside of a bend (or curved streamline) and lower at the inside due to centrifugal forces
- Pressure increases with depth in a liquid due to gravity
- Heat flows from regions of high to low temperature
- Hot fluid rises and cold fluid sinks under the influence of gravity
- Turbulence is generated in regions with sheared flow, i.e. where velocity gradients are high

It is obviously not possible to give a comprehensive list of items, and we should aim to develop specific checks for the flow problem to be studied based on our knowledge of fluid dynamics, heat transfer etc. and comprehensive research of the background to the problem.

10.8 Summary

Trust and confidence are essential issues in industrial applications of CFD modelling. In this chapter we have defined:

- Errors: deficiencies in a CFD model that are not caused by lack of knowledge
- Uncertainty: deficiencies in a CFD model that are caused by lack of knowledge

The main sources of errors are:

- Numerical errors: roundoff, truncation of iterative sequences, discretisation error
- Coding errors
- User errors

In our discussion of iterative convergence we have introduced the definition of the most commonly used convergence indicator: the normalised global residual. We have also touched on the importance of good mesh design for the control of the level and distribution of discretisation errors.

The main sources of uncertainty are:

- Input uncertainty: model deficiencies associated with limited or inaccurate knowledge of domain geometry, boundary conditions or fluid properties
- Physical model uncertainty: model deficiencies due to limited accuracy or lack of validity of submodels or simplifying assumptions

We have given examples of each of these sources of modelling error and uncertainty and discussed their effect on CFD results. To quantify the errors and uncertainty in CFD results we have defined the processes of verification and validation:

- Verification: the process of determining the match between the CFD results and the conceptual model of the fluid flow to quantify errors
- Validation: the process of determining the match between the CFD results and the real flow problem to quantify uncertainty

Given the dominant contribution of discretisation errors in many practical CFD simulations we have stressed the important role played by systematic mesh refinement studies in the verification process, resulting in the following estimates of error in a fine and coarse solution:

$$E_{U,1} = \frac{U_2 - U_1}{1 - r^p} \tag{10.7a}$$

$$E_{U,2} = r^p \left(\frac{U_2 - U_1}{1 - r^p} \right) \tag{10.7b}$$

We have also introduced the grid convergence index proposed by Roache:

$$\mathrm{GCI}_U = F_S E_U \tag{10.8}$$

and discussed his proposal to use the actual order $\tilde{p}$ of discretisation error decay, which can be obtained from a grid-refinement study using two refinement levels (i.e. three grids):

$$\tilde{p} = \ln\left(\frac{U_3 - U_2}{U_2 - U_1} \right) \Big/ \ln(r) \tag{10.9}$$

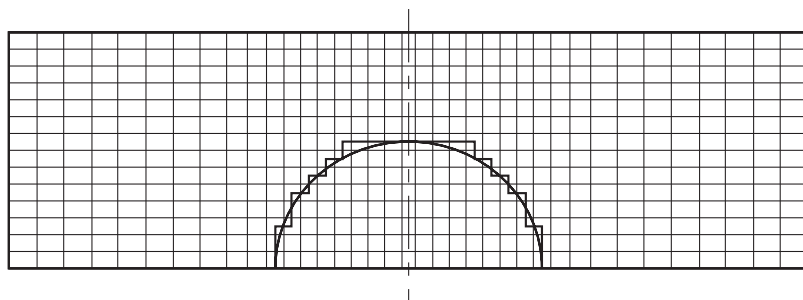
We have outlined the crucial part played by validation and given pointers to trusted, high-quality sources of experimental data that can be used for the validation of CFD codes and models.

Finally, we have summarised brief extracts from the prominent AIAA and ERCOFTAC guidelines for best practice in CFD, which have been developed to help users get the best possible results out of CFD with available computing resources. We have also made some recommendations for uniform reporting and documentation of CFD models and their results.

Chapter eleven Methods for dealing with complex geometries

11.1 Introduction

Techniques of solving fluid flow equations shown in earlier chapters were based on discretisation procedures using the Cartesian co-ordinate system. This is the simplest context, which allowed us to introduce the fundamentals of the finite volume method in a form that is easiest to understand. Extension of the methods developed in Chapters 4 to 6 to other orthogonal co-ordinate systems (cylindrical, axisymmetric three-dimensional or spherical co-ordinates) is relatively straightforward, provided that we write down the governing equations using the appropriate form of the div and grad operators for the chosen co-ordinate system (see Bird *et al*. (2002) for relevant operator definitions). However, many engineering problems involve complex geometries that do not fit exactly in Cartesian co-ordinates or one of the other systems. When the flow boundary does not coincide with the co-ordinate lines of a structured grid, we could proceed by approximating the geometry. This is illustrated in Figure 11.1, where we consider a two-dimensional calculation of the flow past a half cylinder.

Figure 11.1 Cartesian grid arrangement for the prediction of flow over a half cylinder

The only way to represent the curved surface of the half cylinder in a Cartesian co-ordinate system is to use a stepwise approximation. Such an approximate boundary description is tedious and time consuming to set up. Moreover, the cells inside the solid part of the cylinder do not take part in the calculations, so they need to be blocked out, which represents a waste of computer storage and resources. Finally, the stepwise representation of the smooth cylinder wall introduces errors in the computation of wall shear stresses, heat fluxes etc. These errors can be reduced by introducing a very fine Cartesian mesh to cover the wall region, but the structure of grid lines causes further wastage of computer storage due to unnecessary refinement in interior regions where this is of minimal interest.

This example clearly shows that CFD methods based on Cartesian or cylindrical co-ordinate systems have limitations in irregular geometries. Practically important flows with complex geometry are plentiful and include building configurations, furnaces, modern pent-roof combustion chambers in internal combustion (IC) engines, intake and exhaust ports and flow passages, flow past aerofoils, gas turbine combustors, turbomachinery and many more. In such cases it would obviously be much more advantageous to work with grids that can handle curvature and geometric complexity more naturally.

CFD methods for complex geometries are classified into two groups: (i) structured curvilinear grid arrangements and (ii) unstructured grid arrangements. A Cartesian grid is an example of a structured method. In a structured grid arrangement:

- Grid points are placed at the intersections of co-ordinates lines
- Interior grid points have a fixed number of neighbouring grid points
- Grid points can be mapped into a matrix; their location in the grid structure and in the matrix is given by indices (I, J in two dimensions and I, J, K in three dimensions)

Structured curvilinear grids or **body-fitted grids** are based on mapping of the flow domain onto a computational domain with a simple shape. These techniques can deal effectively with flows such as the above half-cylinder problem. Unfortunately, it proves to be quite difficult to find viable mappings when the geometry becomes very complex. In these cases it is often advantageous to be able to sub-divide the flow domain into several different sub-regions or blocks, each of which is meshed separately and joined up correctly with its neighbours. This leads to so-called **block-structured grids**, which are considerably more flexible than Cartesian or body-fitted meshes. The basics of body-fitted and block-structured grid methods are summarised in sections 11.2–11.5.

For the most complex geometries it may be necessary to use many blocks, and the logical extension of this idea is the **unstructured grid**, where each mesh cell is a block. This gives unlimited geometric flexibility and allows the most efficient use of computing resources for complex flows, so this technique is now widely used in industrial CFD. We examine the main elements of the finite volume method for unstructured grids in some more detail in sections 11.6–11.11.

11.2 Body-fitted co-ordinate grids for complex geometries

Methods based on body-fitted grid systems have been developed to deal with curved boundary flows such as the flow over an aerofoil (Rhie and Chow, 1983; Peric, 1985; Demirdzic *et al.*, 1987; Shyy *et al.*, 1988; Karki and Patankar, 1988). There are two types of body-fitted co-ordinate system: (i) orthogonal curvilinear co-ordinates and (ii) non-orthogonal co-ordinates. In an orthogonal mesh the grid lines are perpendicular at intersections. Figure 11.2 shows an example of an **orthogonal curvilinear mesh** for the calculation of flow around an aerofoil.

In Figure 11.3 we present a **non-orthogonal body-fitted grid** for the half-cylinder problem mentioned above. Here the grid lines do not intersect at 90° angles. In both types of body-fitted grid all the domain boundaries coincide with co-ordinate lines, so geometrical details can be incorporated accurately without the need for stepwise approximations. Furthermore, as

Figure 11.2 An example of an orthogonal curvilinear mesh for calculating flow around an aerofoil
Source: Haselbacher (1999)

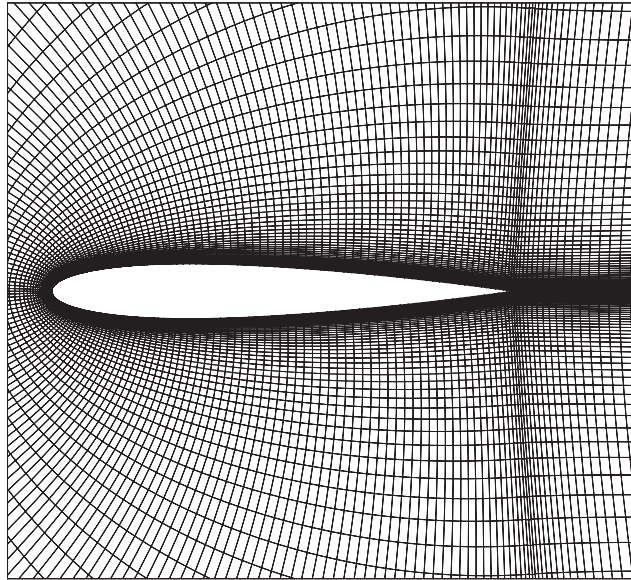



Figure 11.3 Use of a non-orthogonal body-fitted grid arrangement for the prediction of flow over a cylinder

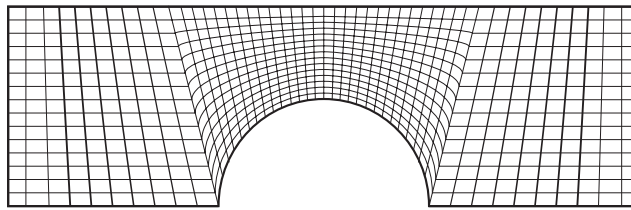



Figure 11.2 demonstrates, the grid can be refined easily to capture important flow features, e.g. in regions with large gradients such as boundary layers.

11.3 Cartesian vs. curvilinear grids – an example

Figure 11.4 shows part of a heat exchanger tube bank where CFD can be used to predict the flow field. Considering symmetry, only the shaded region of the geometry needs to be considered. Figure 11.5a shows a Cartesian grid arrangement to predict this flow. We use a 40×15 mesh, block the cylinder off with dead solid wall cells that do not take part in the calculation, and approximate the surface by means of steps. Figure 11.6a illustrates a non-orthogonal body-fitted grid for the same problem with the same number of cells (i.e. 40×15). Now the whole grid occupies the computational domain and the cylinder surfaces can be more accurately represented. Comparison of Figures 11.5a and 11.6a confirms that only about 75% of the cells in the Cartesian grid are available to represent the flow region; the remaining 25% are wasted in dealing with the objects.

The resulting velocity predictions are shown in Figures 11.5b and 11.6b, respectively. The latter shows much improved definition of the flow in the regions with large curvature near the inlet and outlet. This clearly demonstrates the advantage of the body-fitted grid: computational resources are utilised more efficiently, so flow details can be captured with coarser grids compared with Cartesian-based methods (see Peric, 1985; Rodi *et al.*, 1989).

Figure 11.4 Flow over a heat exchanger tube bank (only a part shown)

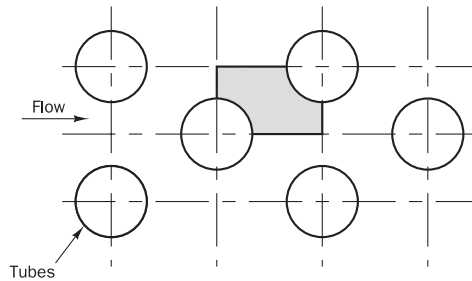



Figure 11.5 (a) Cartesian grid using an approximated profile to represent cylindrical surfaces; (b) predicted flow pattern using a 40×15 Cartesian grid

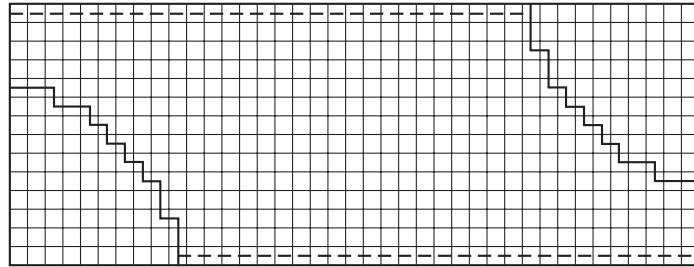
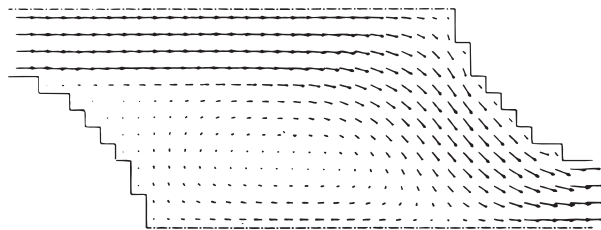
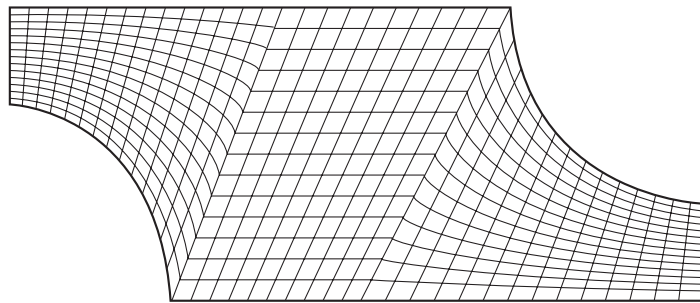
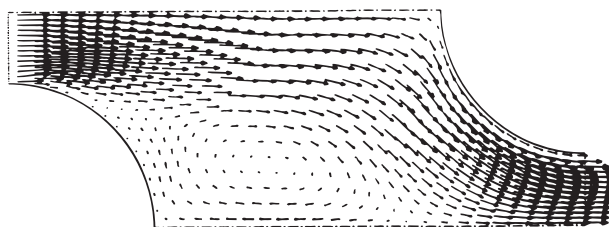



Figure 11.6 (a) Non-orthogonal body-fitted grid for the same problem; (b) predicted flow pattern using a 40×15 structured body-fitted grid

11.4 Curvilinear grids – difficulties

Body-fitted grids have significant advantages over their Cartesian equivalents, but there is a price to pay for the geometric flexibility: the governing equations in curvilinear co-ordinate systems are much more complex. Detailed discussions of the available methods of formulating the governing equations can be found in Demirdzic (1982), Shyy and Vu (1991) and Ferziger and Peric (2001). The main difference between the different formulations lies in the grid arrangement and in the choice of dependent variables in the momentum equations. In CFD procedures based on body-fitted co-ordinates the use of non-staggered or co-located grid systems for velocities is increasingly preferred to staggered grids, which require additional storage allocations. However, special procedures are needed for non-staggered grids to ensure proper velocity and pressure coupling and prevent the occurrence of checker-board pressure oscillations identified in section 6.2. Unstructured grids also use these co-located grid arrangements, and we discuss them further in section 11.14.

In addition to the greater complexity of the equations, it should be noted that body-fitted grids are still structured, so grid refinement is generally not purely local. For example, in Figure 11.2 the refinement needed to resolve the boundary layers and trailing edge geometry persists elsewhere in the interior mesh. This shows up as regions of increased mesh density above, below and downstream from the aerofoil roughly along three lines that originate from the trailing edge. The number of mesh cells in the downstream direction is particularly large, which represents a waste of computer storage.

Use of orthogonal and non-orthogonal body-fitted grids allows us to capture the geometric details, but there can be difficulties associated with their creation. To generate meshes that include all the geometrical details, it is necessary to map the physical geometry into a computational geometry. Mathematical details of the mapping process are not presented here; the interested user should consult the relevant literature for details (see Thomson, 1984, 1988). An example of the mapping process for a part of a tube bank is shown in Figures 11.7a–b. For this comparatively simple geometry it is straightforward to develop a viable mapping, but when the domain geometry is more complex and/or involves a large number of internal objects this can be a very tedious task.

Figures 11.8a–b illustrate the difficulties of generating a body-fitted grid for a pent-roof IC engine combustion chamber by mapping the cylinder geometry into a single three-dimensional hexahedral block (Henson, 1998). Valve details were created by carefully mapping the circular valves to square regions. In addition, the grid had to accommodate piston bowl details, shown on the surface mesh of Figure 11.8a. Various smoothing techniques were used to improve the grid distributions, but the final grid still contains regions with very acute angles and cells with undesirable aspect ratios, even after smoothing. The four regions with dense surface mesh are the result of the need to accommodate valve and pent-roof details. These groups of highly skewed cells can lead to stability problems for CFD solvers. Such bad regions in a mesh may have to be manually adjusted.

Therefore, in spite of their undoubted advantages over simple Cartesian grids, the following problems are encountered with general orthogonal and non-orthogonal **structured** grids:

Figure 11.7 Mapping of physical geometry to computational geometry in structured meshes: (a) physical grid in x, y co-ordinates; (b) the mapped structure for (a) in the computational domain

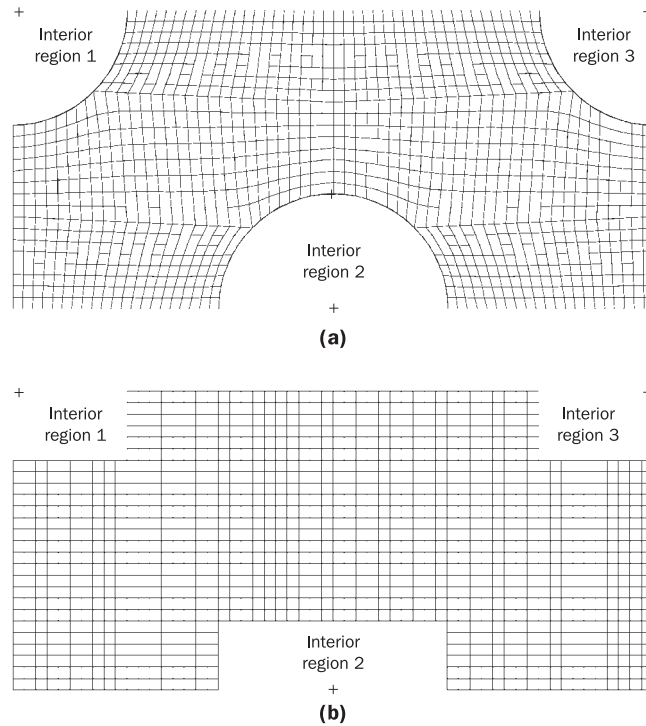
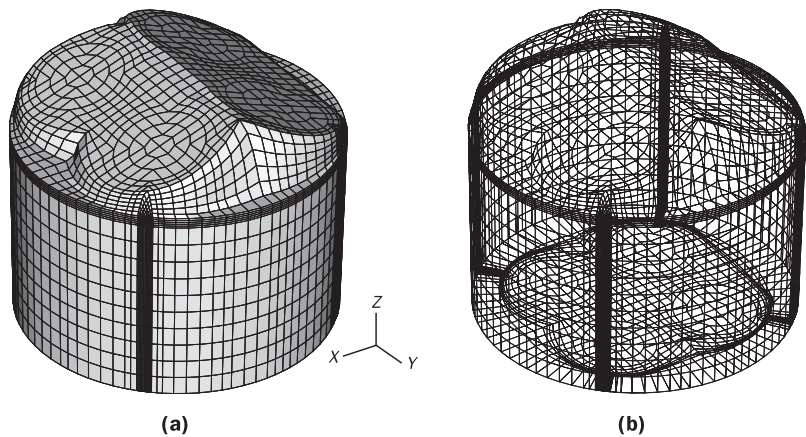



Figure 11.8 A structured non-orthogonal mesh for a pent-roof i.c. engine geometry

- Still difficult and time consuming to generate
- If the solution domain cannot be readily mapped into a rectangle (in 2D) or rectangular parallelepiped (in 3D) this can result in skewed grid lines causing unnecessary local variations
- Unnecessary grid resolutions can result in cases where mapping is difficult
- Mapping is sometime impossible with complex 3D geometries with internal objects/parts

11.5 Block-structured grids

To overcome the problems associated with structured grid generation for complex geometries, block-structured CFD methods have been developed. In a **block-structured grid**, the domain is sub-divided into regions, each of which has a structured mesh. The mesh structure in each region can be different, and it is even possible to use different co-ordinate systems. Such meshes are more flexible than ('single block') structured meshes described in the previous sections. The block-structured approach allows the use of fine grids in regions where greater resolution is required. The interfaces of adjacent blocks may have grids on either side that are matching or non-matching, but, either way, they must be properly treated in a fully conservative manner. In some codes the solvers are applied in a block-wise manner (block by block with overall final iterations to unify boundary conditions) and local refinement is possible block-wise. Block-structured grids with overlapping regions are called composite grids or chimera grids. Figure 11.9 shows a Cartesian block-structured grid used for the calculation of flow over an aerofoil. The resulting grid structure combines the advantages of Cartesian grids – easy to generate, equations simple to discretise and solve – with the ability of curvilinear grids to accommodate curved complex boundaries (see Courier and Powell, 1996).

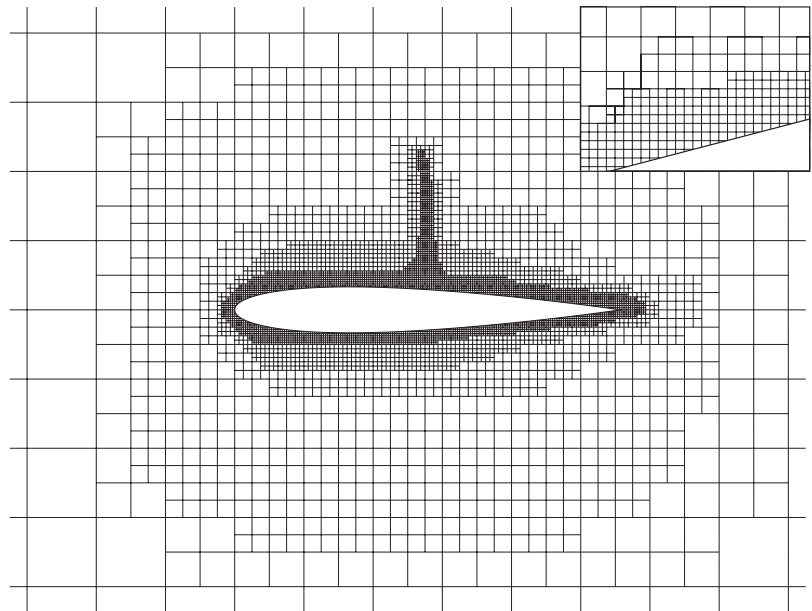
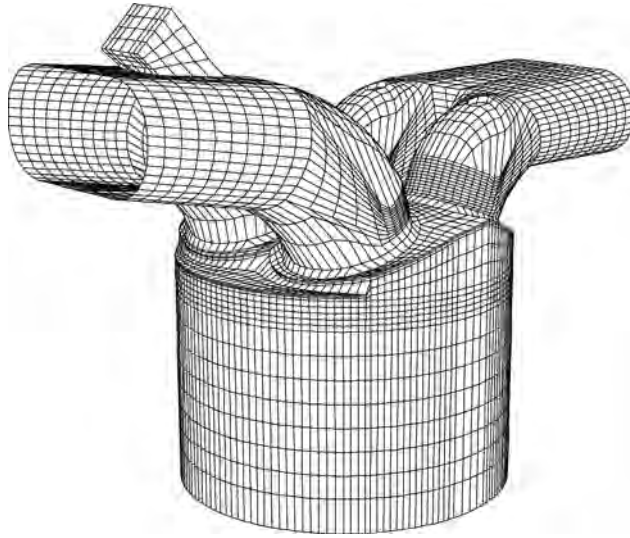
Figure 11.9 Block-structured mesh for a transonic aerofoil. Inset shows cut cells near aerofoil surface. Also note additional grid refinement in the flow region to capture a shock above the aerofoil
Source: Haselbacher (1999)

Block-structured meshes are extremely useful in handling complex geometries that consist of several geometrical sub-components such as the IC engine pent-roof cylinder and inlet port geometry. Figure 11.10 demonstrates the improvement of grid quality that was achieved by applying the block-structured meshing in the engine code KIVA-3V to define separate blocks for mesh inlet ports, valve regions and the engine cylinder (generated using the pre-processor of Kiva 3V: see Amsden, 1997).

Figure 11.10 Block-structured mesh arrangement for an engine geometry, including inlet and exhaust ports, used in engine simulations with KIVA-3V

11.6 Unstructured grids

An **unstructured grid** can be thought of as a limiting case of a multi-block grid where each individual cell is treated as a block. The advantage of such an arrangement is that no implicit structure of co-ordinate lines is imposed by the grid – hence the name unstructured – and the mesh can be easily concentrated where necessary without wasting computer storage. Moreover, control volumes may have any shape, and there are no restrictions on the number of adjacent cells meeting at a point (2D) or along a line (3D). In practical CFD, triangles or quadrilaterals are most often used for 2D problems and tetrahedral or hexahedral elements in 3D ones. Figure 11.11 shows a triangular unstructured mesh for the calculation of a 2D flow over an aerofoil.

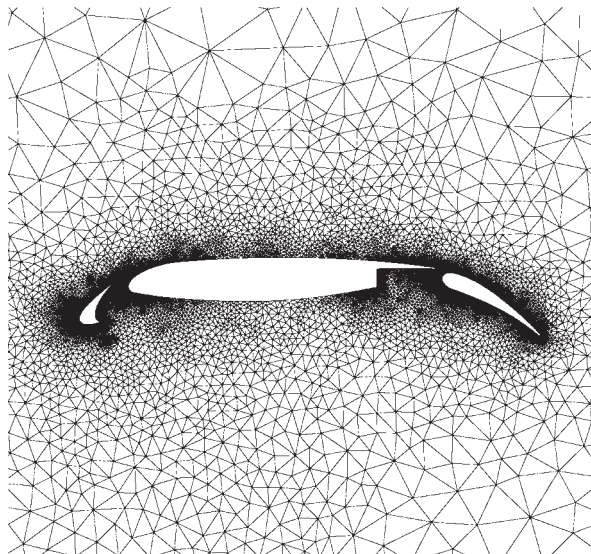
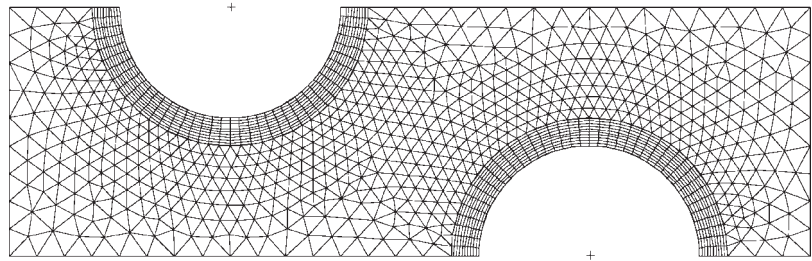
Figure 11.11 A triangular grid for a three-element aerofoil
Source: Haselbacher (1999)

In unstructured grid arrangements we are not restricted to one particular cell type, but it is possible to use a mixture of cell shapes. In 2D a mixture of triangular and quadrilateral elements can be used to construct the grid. In 3D flow calculations combinations of tetrahedral and hexahedral elements are frequently used. Such grids are called hybrid meshes. Figure 11.12 shows an example of a hybrid unstructured grid for the calculation of flow in a tube bank where quadrilateral cells have been used near solid walls to provide better resolution of the viscous effects in the boundary layers and an expanding triangular mesh structure elsewhere to utilise the resources efficiently.

Figure 11.12 An example of an unstructured mesh with mixed elements

The most attractive feature of the unstructured mesh is that it allows the calculation of flows in or around geometrical features of arbitrary complexity without having to spend a long time on mesh generation and mapping. Grid generation is fairly straightforward (especially with triangular and tetrahedral grids), and automatic generation techniques, originally developed for finite element methods, are now widely available. Furthermore, mesh refinement and adaption (semi-automatic mesh refinement to improve resolution in regions with large gradients) are much easier in unstructured meshes. In the sections to follow we explore the unstructured methodology in more detail as it is now the most popular technique and is included in all commercial CFD codes on the market today.

11.7 Discretisation in unstructured grids

Unstructured grids are the most general form of grid arrangement for most complex geometries. Here we present a brief outline of discretisation techniques for unstructured grids with arbitrary cell shapes, which may be bounded by any number of control surfaces. We limit ourselves to the development of the main ideas; interested readers should consult the literature for further details of the methodology.

There are two ways of defining control volumes in unstructured meshes: (i) cell-centred control volumes and (ii) vertex-centred control volumes. These two variants are illustrated in Figure 11.13 for a 2D problem.

In the **cell-centred method** the nodes are placed at the centroid of the control volume, as shown in Figure 11.13a. In the **vertex-centred method** the nodes are placed on the vertices of the grid. This is followed by a process known as median-dual tessellation, whereby sub-volumes are formed by joining centroids of the elements and midpoints of the edges, as illustrated in

Figure 11.13 Control volume construction in 2D unstructured meshes: (a) cell-centred control volumes; (b) vertex-based control volumes

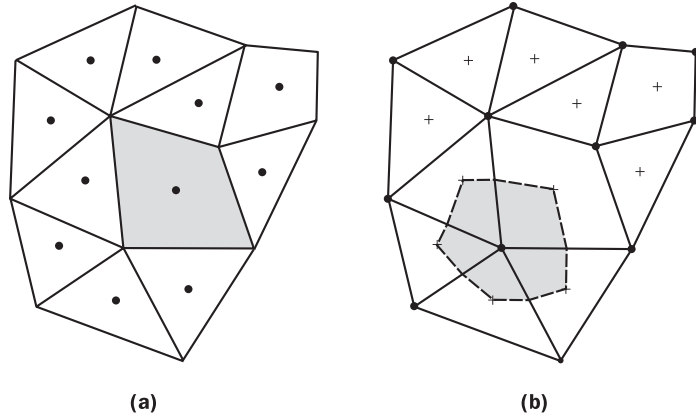



Figure 11.13b. The sub-volume surrounding a node then forms the control volume for discretisation. Both cell-centred and vertex-centred methods are used in practice. We develop the ideas of discretisation in unstructured grids for the cell-centred method, which is simpler to understand, and, since a control volume always has more vertices than centroids, it has slightly lower storage requirements than the vertex-centred method.

The discretisation in unstructured meshes can be developed from the basic control volume technique introduced in earlier chapters, where we used the integral form (2.40) of the conservation equation as the starting point:

$$\int_{CV} \frac{\partial}{\partial t}(\rho\phi)\mathrm{d}V + \int_{CV} \mathrm{div}(\rho\phi\mathbf{u})\mathrm{d}V = \int_{CV} \mathrm{div}(\Gamma\ \mathrm{grad}\ \phi)\mathrm{d}V + \int_{CV} S_\phi \mathrm{d}V \qquad (11.1)$$

The volume integration in the transient term on the left hand side and the source term on the right hand side can be conveniently evaluated as the product of the volume of the cell and the relevant centroid value of the integrand. The time integration can be treated using the explicit or implicit techniques developed in Chapter 8.

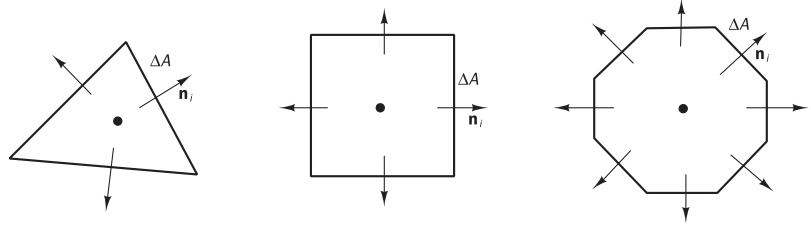
Equation (11.1) also contains terms with the divergence of the convective flux ($\rho\phi\mathbf{u}$) and of the diffusive flux (Γ grad ϕ). In the absence of a specific co-ordinate system these terms need careful treatment. We recall Gauss's theorem (2.41), which is applicable to any shape of control volume:

$$\int_{CV} \mathrm{div}\ \mathbf{a}\ \mathrm{d}V = \int_{A} \mathbf{n} \cdot \mathbf{a}\ \mathrm{d}A \qquad (11.2)$$

The surface integration must be carried out over the bounding surface A of the control volume CV. The physical interpretation of $\mathbf{n} \cdot \mathbf{a}$ is the component of the vector $\mathbf{a}$ in the direction of the outward unit vector $\mathbf{n}$ normal to infinitesimal surface element dA.

Some simple 2D examples of different shapes of control volumes are shown in Figure 11.14. We note that the bounding surface or control surface of each 2D control volume is a closed contour formed by means of a series of finite-sized straight line elements, the area of which is denoted by ΔA. In 3D

the control volume would be bounded by triangular or quadrilateral surface elements.

Figure 11.14 Typical 2D control volumes with varying number of surface elements

Application of Gauss's theorem to equation (11.1) gives

$$\frac{\partial}{\partial t}\left(\int_{CV} \rho\phi \mathrm{d}V\right) + \int_A \mathbf{n} \,.\, (\rho\phi\mathbf{u})\mathrm{d}A = \int_A \mathbf{n} \,.\, (\Gamma \text{ grad } \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \qquad (11.3)$$

Note that A is the area of the entire control surface in equation (11.3) and dA indicates an infinitesimal surface element. The area integrations are carried out over all line segments (2D) or surface elements (3D), so they can be written as follows:

$$\frac{\partial}{\partial t}\left(\int_{CV} \rho\phi \mathrm{d}V\right) + \sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \,.\, (\rho\phi\mathbf{u})\mathrm{d}A$$

$$= \sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \,.\, (\Gamma \text{ grad } \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \qquad (11.4)$$

For steady flows we have

$$\int_A \mathbf{n} \,.\, (\rho\phi\mathbf{u})\mathrm{d}A = \int_A \mathbf{n} \,.\, (\Gamma \text{ grad } \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \qquad (11.5)$$

and hence

$$\sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \,.\, (\rho\phi\mathbf{u})\mathrm{d}A = \sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \,.\, (\Gamma \text{ grad } \phi)\mathrm{d}A + \int_{CV} S_\phi \mathrm{d}V \qquad (11.6)$$

To evaluate the control surface integrations we need expressions for flux vectors $(\rho\phi\mathbf{u})$ and (Γ grad ϕ) as well as geometric quantities $\mathbf{n}_i$ and ΔA_i. In sections 11.7 and 11.8 we develop special expressions for the diffusive flux $\mathbf{n}_i$. (Γ grad ϕ) and convective flux $\mathbf{n}_i$. $(\rho\phi\mathbf{u})$ across line segments or surface elements. Here we show how the outward normal vector $\mathbf{n}_i$ and surface element area ΔA_i can be calculated using simple trigonometry and vector algebra from the vertex co-ordinates of the unstructured grid.

A typical cell-centred arrangement is shown in Figure 11.15 along with the notations we will use to describe the discretisation procedure.

Figure 11.15 Cell-centred control volume arrangement

In this figure point P is the centroid of the control volume for which we develop the discretisation process. Point A is the centroid of the adjacent control volume and $\mathbf{e}_\xi$ is a unit vector along the line joining P and A. The face separating the two control volumes is identified as 'i', and ab is a line joining vertices a and b, which are shared by the two control volumes. The co-ordinates of points a and b are (x_a, y_a) and (x_b, y_b) respectively. Unit vectors $\mathbf{n}$ and $\mathbf{e}_\eta$ are, respectively, the outward normal vector and tangent vector to face i.

We now calculate the required geometry parameters for equations (11.4) and (11.6) as follows. Consider the control volume face shown in Figure 11.16.

Figure 11.16 A face of a control volume and the normal unit vector

The area of the face is given by

$$\Delta A_i = \sqrt{(\Delta x)^2 + (\Delta y)^2}$$

where $\Delta x = x_b - x_a$ and $\Delta y = y_b - y_a$

The normal unit vector to the surface is defined by

$$\mathbf{n} = \frac{\Delta y}{\Delta A_i}\mathbf{i} - \frac{\Delta x}{\Delta A_i}\mathbf{j} \tag{11.7}$$

In the absence of a grid structure it is necessary to create a data structure for the geometry information along with a method of identifying the relationship between vertices, cell indices, relevant edges and neighbouring cell indices.

11.8 Discretisation of the diffusion term

In equations (11.4) and (11.6) the diffusion term has been written as a sum over all the surface elements that make up the bounding surface of a control volume:

$$\sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \,.\, (\Gamma \text{ grad } \phi)\, \mathrm{d}A$$

The area integration for each of the elements is approximated by the dot product of the outward unit normal vector $\mathbf{n}_i$ and a representative diffusive flux vector (Γ grad ϕ) for the control surface element ΔA_i. The latter can be approximated easily using the **central differencing** method along line PA. Thus,

$$\int_{\Delta A_i} \mathbf{n}_i \,.\, (\Gamma \text{ grad } \phi)\mathrm{d}A \cong \mathbf{n}_i \,.\, (\Gamma \text{ grad } \phi)\Delta A_i \cong \Gamma\left(\frac{\phi_A - \phi_P}{\Delta\xi}\right)\Delta A_i \tag{11.8}$$

In equation (11.8) $\Delta\xi$ is the distance between the centroids A and P. It should be noted that central difference (11.8) is only accurate if the line joining nodes P and A and the unit normal vector $\mathbf{n}_i$ are in the same direction, so the approximation is only correct if the mesh is fully orthogonal. Generally, in unstructured meshes the lines connecting centroids P and A are *not* parallel to the unit normal vector $\mathbf{n}_i$, as shown in Figure 11.15. This is known as mesh skewness or non-orthogonality. The flux calculation (11.8) therefore has to be corrected by adding a contribution arising from non-orthogonality. There are different ways to correct the flux (e.g. Davidson, 1996; Mathur and Murthy, 1997; Haselbacher, 1999; Kim and Choi, 2000; Ferziger and Peric, 2001), but the most common form is to introduce a term known as **cross-diffusion**, which is treated as a source term when the discretised equation is assembled.

We follow Mathur and Murthy (1997) and develop an expression for the cross-diffusion term by introducing co-ordinates ξ along the line joining P and A, and η along the face of the control volume (i.e. along the line joining vertices a and b). Figure 11.17a shows that the outward unit normal vector $\mathbf{n}_i$ is perpendicular to the tangential co-ordinate η. Thus, the term grad ϕ can be expressed in terms of x, y coordinates or n, η coordinates as follows:

$$\text{grad } \phi = \frac{\partial\phi}{\partial x}\mathbf{i} + \frac{\partial\phi}{\partial y}\mathbf{j} = \frac{\partial\phi}{\partial n}\mathbf{n} + \frac{\partial\phi}{\partial\eta}\mathbf{e}_\eta \tag{11.9}$$

where $\mathbf{n}$ and $\mathbf{e}_\eta$ are unit vectors along normal and tangential directions.

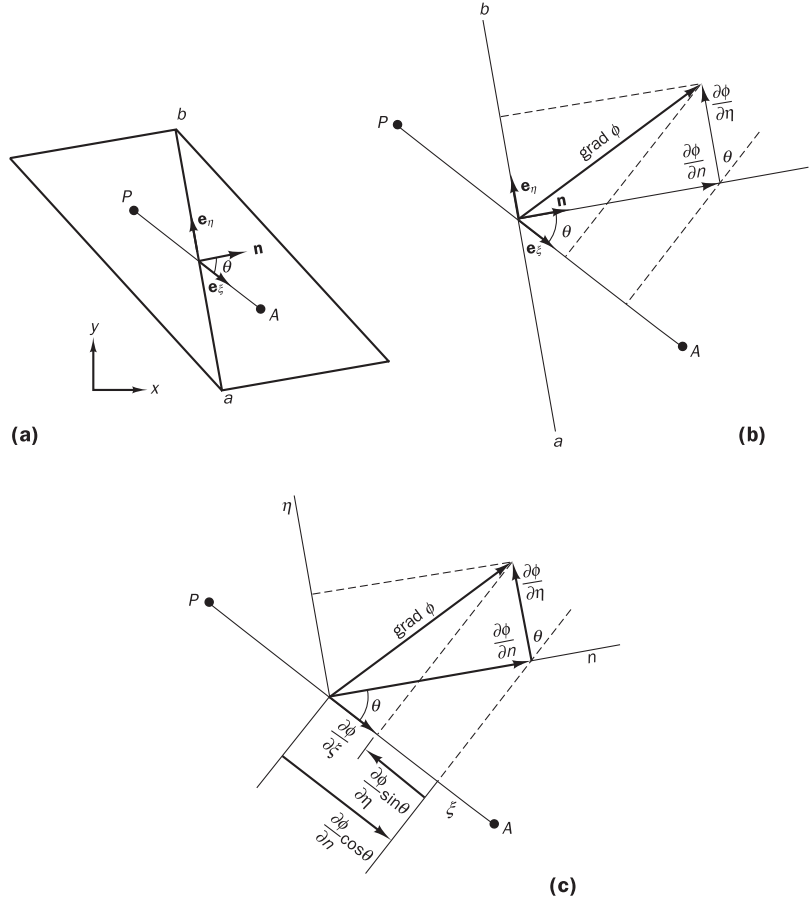
As an aside we note that the normal unit vector $\mathbf{n}$ and the two other unit vectors $\mathbf{e}_\xi$ and $\mathbf{e}_\eta$ in the directions of ξ and η, respectively, can be calculated from stored x- and y-co-ordinates of control volume nodes and vertices as follows (see Figures 11.16 and 11.17a):

$$\mathbf{n} = \frac{\Delta y}{\Delta A_i}\mathbf{i} - \frac{\Delta x}{\Delta A_i}\mathbf{j} = \frac{y_b - y_a}{\Delta\eta}\mathbf{i} - \frac{x_b - x_a}{\Delta\eta}\mathbf{j} \tag{11.10}$$

$$\mathbf{e}_\xi = \frac{x_A - x_P}{\Delta\xi}\mathbf{i} + \frac{y_A - y_P}{\Delta\xi}\mathbf{j} \tag{11.11}$$

$$\mathbf{e}_\eta = \frac{x_b - x_a}{\Delta\eta}\mathbf{i} + \frac{y_b - y_a}{\Delta\eta}\mathbf{j} \tag{11.12}$$

Before developing an expression for the cross-diffusion term, we note that the central difference on the right hand side of Equation (11.8) is of course only actually an approximation of $\partial\phi/\partial\xi$, whereas the left hand side actually requires $\mathbf{n}$. grad $\phi = \partial\phi/\partial n$. If the mesh is orthogonal $\partial\phi/\partial\xi = \partial\phi/\partial n$ and the central difference approximation is correct, but if the mesh is non-orthogonal $\partial\phi/\partial\xi$ may be very different from $\partial\phi/\partial n$.

Figure 11.17 Definition sketch for evaluation of cross-diffusion term

Figures 11.17b–c show that $\partial\phi/\partial\xi$ corresponds to the length of the projection of vector grad ϕ in the direction of ξ. Using the expression (11.9) we can also represent grad ϕ as the vector sum of the two components $(\partial\phi/\partial n)\mathbf{n}$ and $(\partial\phi/\partial\eta)\mathbf{e}_\eta$, as shown in Figure 11.17b. To get an improved estimate for the normal flux $\mathbf{n}$. grad $\phi = \partial\phi/\partial n$ we examine the relationship between the projection of grad ϕ in the direction of ξ, i.e. $\partial\phi/\partial\xi$, and the projections in that direction of the two components $(\partial\phi/\partial n)\mathbf{n} \cdot \mathbf{e}_\xi$ and $(\partial\phi/\partial\eta)\mathbf{e}_\eta \cdot \mathbf{e}_\xi$ of grad ϕ. Figure 11.17 illustrates how the lengths of the projections of the two component vectors can be calculated if the angle between the $\mathbf{n}$- and ξ-directions is denoted by θ:

$$\frac{\partial\phi}{\partial n}\mathbf{n} \cdot \mathbf{e}_\xi = \frac{\partial\phi}{\partial n}\cos(\theta) \tag{11.13}$$

and

$$\frac{\partial \phi}{\partial \eta}\mathbf{e}_\eta \cdot \mathbf{e}_\xi = -\frac{\partial \phi}{\partial \eta}\sin(\theta) \tag{11.14}$$

The magnitude of the component of grad ϕ in the direction of ξ is just $\partial\phi/\partial\xi$, which is also equal to the sum of the two projections (11.13) and (11.14). Hence,

$$\frac{\partial \phi}{\partial \xi} = \frac{\partial \phi}{\partial n}\cos(\theta) - \frac{\partial \phi}{\partial \eta}\sin(\theta) \tag{11.15}$$

We remember that $\mathbf{n}$. grad $\phi = \partial\phi/\partial n$ and rearrange (11.15) to obtain the following expression for the normal component of the diffusive flux required in equation (11.8):

$$\mathbf{n} \cdot \text{grad}\ \phi = \frac{\partial \phi}{\partial n} = \frac{\partial \phi}{\partial \xi}\frac{1}{\cos(\theta)} + \frac{\partial \phi}{\partial \eta}\tan(\theta) \tag{11.16}$$

The two gradients of the transported quantity ϕ on the right hand side of expression (11.16) may both be approximated using central differencing:

$$\frac{\partial \phi}{\partial \xi} = \frac{\phi_A - \phi_P}{\Delta\xi} \tag{11.17}$$

$$\frac{\partial \phi}{\partial \eta} = \frac{\phi_b - \phi_a}{\Delta\eta} \tag{11.18}$$

where $\Delta\xi = d_{PA}$ is the distance between points A and P
and $\Delta\eta = d_{ab}$ is the distance between vertices a and b ($= \Delta A_i$)

In the literature $\partial\phi/\partial\xi$ and $\partial\phi/\partial\eta$ are called the direct gradient and cross-diffusion, respectively. Substitution of central difference approximations (11.17) and (11.18) into equation (11.16) yields

$$\mathbf{n} \cdot \text{grad}\ \phi\ \Delta A_i = \frac{\Delta A_i}{\cos(\theta)}\frac{\phi_A - \phi_P}{\Delta\xi} + \Delta A_i \tan(\theta)\frac{\phi_b - \phi_a}{\Delta\eta} \tag{11.19}$$

It is straightforward to see in Figure 11.17 that

$$\frac{1}{\cos(\theta)} = \frac{1}{\mathbf{n} \cdot \mathbf{e}_\xi} = \frac{\mathbf{n} \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{e}_\xi} \tag{11.20}$$

and

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} = -\frac{\mathbf{e}_\xi \cdot \mathbf{e}_\eta}{\mathbf{n} \cdot \mathbf{e}_\xi} \tag{11.21}$$

Thus, (11.19) can be written in vector form as follows:

$$\mathbf{n} \cdot \text{grad}\ \phi\ \Delta A_i = \underbrace{\frac{\mathbf{n} \cdot \mathbf{n}\Delta A_i}{\mathbf{n} \cdot \mathbf{e}_\xi}\frac{\phi_A - \phi_P}{\Delta\xi}}_{\text{Direct gradient term}} - \underbrace{\frac{\mathbf{e}_\xi \cdot \mathbf{e}_\eta \Delta A_i}{\mathbf{n} \cdot \mathbf{e}_\xi}\frac{\phi_b - \phi_a}{\Delta\eta}}_{\text{Cross-diffusion term}} \tag{11.22}$$

The factors $\mathbf{n} \cdot \mathbf{n}\Delta A_i/\mathbf{n} \cdot \mathbf{e}_\xi$ and $\mathbf{e}_\xi \cdot \mathbf{e}_\eta \Delta A_i/\mathbf{n} \cdot \mathbf{e}_\xi$ can be calculated from the grid geometry. An alternative derivation to obtain equation (11.22) is presented in Appendix F.

Usually the cross-diffusion term is treated as a source term in the discretised form. Therefore separating the cross-diffusion term equation (11.22) is written as

$$\mathbf{n} \cdot \text{grad}\ \phi\ \Delta A_i = \frac{\mathbf{n} \cdot \mathbf{n} \Delta A_i}{\mathbf{n} \cdot \mathbf{e}_\xi} \frac{\phi_A - \phi_P}{\Delta \xi} + S_{D\text{-}cross} \tag{11.23}$$

To evaluate the cross-diffusion term the gradient of ϕ along the line ab is required. There are number of methods used for this calculation. One possibility is to interpolate nodal values of ϕ to calculate ϕ_a and ϕ_b and use them to calculate the gradient. Simple averaging over neighbouring nodes would lead to

$$\phi_a = \frac{\phi_P + \phi_A + \phi_B + \dots}{N} \tag{11.24}$$

where N is the number of nodes surrounding the vertex a. Alternatively a distance-weighted average may be used, which is more accurate but more expensive to compute.

The gradient reconstruction methods described in the next section could also be used to evaluate the gradient at vertices, and then linear interpolation may be used to get the gradient at the face centre.

It can be seen that when the grid is orthogonal the unit vector $\mathbf{e}_\xi$ and the unit normal $\mathbf{n}$ are the same. Moreover, unit vectors $\mathbf{e}_\xi$ and $\mathbf{e}_\eta$ are perpendicular, so their dot product is zero and, hence, the cross-diffusion term in equation (11.22) vanishes. Now, the flux is given by equation (11.8).

In summary, for unstructured grids the **diffusion flux** through each control volume face is evaluated as follows:

$$\boxed{\mathbf{n} \cdot \Gamma\ \text{grad}\ \phi\ \Delta A_i = D_i(\phi_A - \phi_P) + S_{D\text{-}cross,i}} \tag{11.25}$$

where

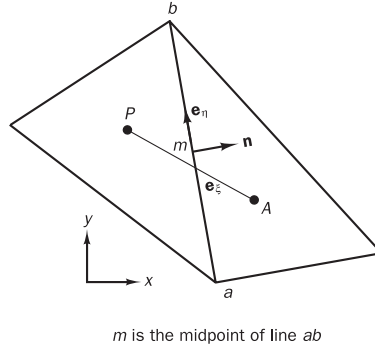
$$\boxed{D_i = \frac{\Gamma}{\Delta \xi} \frac{\mathbf{n} \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{e}_\xi} \Delta A_i}$$

and

$$\boxed{S_{D\text{-}cross,i} = -\Gamma \frac{\mathbf{e}_\xi \cdot \mathbf{e}_\eta \Delta A_i}{\mathbf{n} \cdot \mathbf{e}_\xi} \frac{\phi_b - \phi_a}{\Delta \eta}}$$

It should be noted that the diffusive flux parameter D_i has dimensions (kg/s) of a mass flow rate. This is different from the diffusion conductance D (units kg/m^2.s) that was used in Chapters 4 to 6, since D_i includes the control surface element area ΔA_i.

Figure 11.18 shows that there is a further error term due to the fact that the central differences involved in the control surface element integration are only second-order accurate if they are evaluated using the midpoint value of $\mathbf{n} \cdot \text{grad}\ \phi \Delta A_i$. This is not the case if the lines PA and ab do not intersect at the midpoint m of ab when the grid is non-orthogonal. This error increases with increasing skewness and aspect ratio, so it is important that every effort is made to control skew and aspect ratios in unstructured grids.

Figure 11.18 Geometric sketch for skewed grid with misalignment between midpoint m of line ab and intersection point of lines PA and ab

11.9 Discretisation of the convective term

The convective term of equations (11.4) and (11.6) is

$$\sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \, . \, (\rho\phi\mathbf{u})\, \mathrm{d}A \tag{11.26}$$

The area integral is evaluated as a sum of integrals over all control surface elements ΔA_i. Each of these integrals is approximated by the dot product of the outward unit normal vector $\mathbf{n}_i$ and a representative convective flux vector $(\rho\phi\mathbf{u})$ multiplied by control surface element area ΔA_i. We define convective flux parameter F_i, which is equal to the mass flow rate normal to the surface element:

$$F_i = \int_{\Delta A_i} \mathbf{n}_i \, . \, (\rho\mathbf{u})\, \mathrm{d}A \cong \mathbf{n}_i \, . \, (\rho\mathbf{u})\Delta A_i \tag{11.27}$$

Again we note that the units of the convective flux parameter F_i are those of a mass flow rate (kg/s), in contrast to the dimensions of the convective mass flux per unit area F used throughout Chapters 5 and 6, which has units kg/m^2.s.

The last step in equation (11.27) involves an approximation of the integrand by means of a single representative velocity. A second-order accurate calculation of F_i using a single value is midpoint rule integration, which requires the velocity vector $\mathbf{u}$ at the centre of the face i. In staggered grid arrangements the face velocities are available from the momentum equation and stored at face centres. On the other hand, in co-located grids it is necessary to use interpolated face velocity components for the calculation of mass flux through the face. Special interpolation techniques are employed to overcome the 'checker-board' pressure problem for a co-located arrangement. We postpone discussion of these details until section 11.14.

On the assumption that we somehow have a suitable interpolated value for the face velocity we can write the convective flux of transported quantity ϕ across the control surface in terms of the product as $F_i\phi_i$:

$$\sum_{\text{all surfaces}} \int_{\Delta A_i} \mathbf{n}_i \, . \, (\rho\phi\mathbf{u})\, dA = \sum_{\text{all surfaces}} F_i\phi_i \tag{11.28}$$

where ϕ_i is the value of ϕ at the centre of surface area element i.

As before, we also need to develop methods to generate face centre values ϕ_i of the transported quantity which satisfy the requirements of conservativeness, boundedness and transportiveness that were formulated in Chapter 5. It should be noted that the treatment for general flow variable ϕ is also applicable to velocity components u, v and w without change.

Upwind differencing scheme in unstructured grids

To calculate the convective flux we may utilise the **upwind** approach, which was introduced in section 5.6. The convective flux is $F_i\phi_i$:

For $F_i > 0 \quad \phi_i = \phi_P$
For $F_i < 0 \quad \phi_i = \phi_A$

This is exact if the flow vector **u** is also in the direction of PA (see Figure 11.15). In a general situation the velocity vector may or may not be in the direction of PA. We have also established in earlier discussions that when the flow vector is not in the direction of discretisation (i.e. PA) the upwind scheme gives **false diffusion**. This strongly suggests that we should consider using a higher-order scheme or a TVD scheme for the calculation of the convective flux.

Higher-order differencing schemes in unstructured grids

Recall that in 1D Cartesian grids the **linear upwind differencing** scheme given by equation (5.65) is

$$\phi_e = \phi_P + \left(\frac{\phi_P - \phi_W}{\Delta x}\right)\frac{1}{2}\Delta x$$

where $(\phi_P - \phi_W)/\Delta x$ is the gradient at P and $\Delta x/2$ is distance from P to the face e. The scheme uses an upwind-biased estimate of the gradient at P to calculate the face value $\phi_i = \phi_e$. This can be extended formally to unstructured meshes by using a Taylor series expansion of ϕ about the centroid P:

$$\phi(x, y) = \phi_P + (\nabla\phi)_P \,.\, \Delta\mathbf{r} + O(|\Delta\mathbf{r}|^2) \tag{11.29}$$

where $(\nabla\phi)_P$ is the gradient of ϕ at point P.

If we take $\Delta\mathbf{r}$ as the distance vector from P to the face (see Figure 11.15) then the face value of the transported quantity ϕ can be evaluated by means of

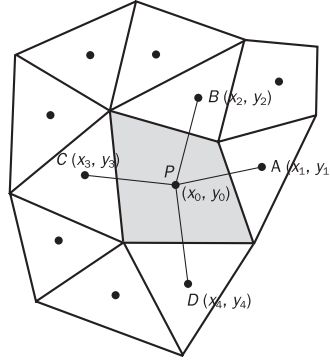
$$\phi_i = \phi_P + (\nabla\phi)_P \,.\, \Delta\mathbf{r} \tag{11.30}$$

Equation (11.29) indicates that the magnitude of the neglected terms is proportional to the square of the distance between node P and the face i, so this is a second-order approximation.

To use equation (11.30) in an unstructured grid to calculate ϕ_i we need $\nabla\phi$ at the point P. In the literature there are several methods available to calculate this quantity. One popular method is to use the so-called **least-squares gradient reconstruction** at P.

Referring to Figure 11.19, the values of the transported quantity ϕ at each node surrounding the centre may be expressed as follows:

$$\phi_i = \phi_0 + \left(\frac{\partial\phi}{\partial x}\right)\bigg|_0 (x_i - x_0) + \left(\frac{\partial\phi}{\partial y}\right)\bigg|_0 (y_i - y_0) \tag{11.31}$$

Figure 11.19 A control volume and its neighbour nodes

Written in another form

$$\phi_i = \phi_0 + \left(\frac{\partial \phi}{\partial x}\right)\bigg|_0 \Delta x_i + \left(\frac{\partial \phi}{\partial y}\right)\bigg|_0 \Delta y_i \tag{11.32}$$

For each node surrounding '0' we have

$$\phi_1 - \phi_0 = \left(\frac{\partial \phi}{\partial x}\right)\bigg|_0 \Delta x_1 + \left(\frac{\partial \phi}{\partial y}\right)\bigg|_0 \Delta y_1 \tag{11.33a}$$

$$\phi_2 - \phi_0 = \left(\frac{\partial \phi}{\partial x}\right)\bigg|_0 \Delta x_2 + \left(\frac{\partial \phi}{\partial y}\right)\bigg|_0 \Delta y_2 \tag{11.33b}$$

$$\phi_3 - \phi_0 = \left(\frac{\partial \phi}{\partial x}\right)\bigg|_0 \Delta x_3 + \left(\frac{\partial \phi}{\partial y}\right)\bigg|_0 \Delta y_3 \tag{11.33c}$$

$$\vdots \qquad\qquad \vdots$$

$$\phi_N - \phi_0 = \left(\frac{\partial \phi}{\partial x}\right)\bigg|_0 \Delta x_N + \left(\frac{\partial \phi}{\partial y}\right)\bigg|_0 \Delta y_N \tag{11.33n}$$

This set of equations can be assembled into a matrix equation as follows:

$$\begin{bmatrix} \Delta x_1 & \Delta y_1 \\ \Delta x_2 & \Delta y_2 \\ \Delta x_3 & \Delta y_3 \\ \vdots & \vdots \\ \Delta x_N & \Delta y_N \end{bmatrix} \begin{bmatrix} \dfrac{\partial \phi}{\partial x}\bigg|_0 \\ \dfrac{\partial \phi}{\partial y}\bigg|_0 \end{bmatrix} = \begin{bmatrix} \phi_1 - \phi_0 \\ \phi_2 - \phi_0 \\ \phi_3 - \phi_0 \\ \vdots \\ \phi_N - \phi_0 \end{bmatrix} \tag{11.34}$$

This represents an overdetermined system of linear equations, in the form $\mathbf{AX} = \mathbf{B}$, which may be solved for $\mathbf{X} = [\partial \phi/\partial x|_0 \; \partial \phi/\partial y|_0]$ using the least-squares approach. Multiplying both sides of the equation by transpose $\mathbf{A}^T$ we obtain

$$\mathbf{A}^T\mathbf{AX} = \mathbf{A}^T\mathbf{B} \tag{11.35}$$

Then $\mathbf{A}^T\mathbf{A}$ becomes a 2×2 matrix that can be easily inverted to solve for $\mathbf{X}$. Since matrix $\mathbf{A}$ depends on geometry only, this calculation needs to be done once for each node. The required gradient vector is obtained from

$$\mathbf{X} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{B} \tag{11.36}$$

Anderson and Bonhaus (1994) commented that this procedure can be very inaccurate when the grid is highly stretched. For such applications these authors recommended the QR decomposition method. Details of the QR decomposition method can be found in Golub and Van Loan (1989). Further details on gradient reconstruction can be found in Haselbacher and Blazek (2000).

TVD schemes in unstructured grids

The concept of TVD schemes for the calculation of convective fluxes was introduced in Chapter 5. As explained there, higher-order schemes such as QUICK can be accommodated in the TVD framework, which is the most general form of discretisation scheme. Darwish and Moukalled (2003) have provided a detailed discussion on the use of TVD schemes in unstructured meshes. Here we summarise their development.

In the usual notation for Cartesian grids it was shown, in section 5.10, that for the positive flow direction, the face value of ϕ using a TVD scheme may be written as

$$\phi_i = \phi_P + \frac{\psi(r)}{2}(\phi_E - \phi_P) \tag{11.37}$$

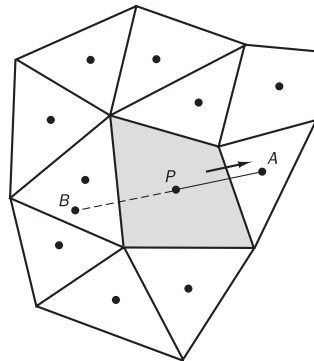
where r is the ratio of the upwind-side gradient to the downwind-side gradient given by

$$r = \frac{\phi_P - \phi_W}{\phi_E - \phi_P} \tag{11.38}$$

Here E is the downstream node and W is the upstream node. In relation to a face of an unstructured cell A is equivalent to E.

However, in unstructured grid arrangements the value of r cannot be written in the same way because the upstream nodal value (assuming flow is positive along P to A) equivalent to W is not available. We need to construct an upstream 'dummy' node B as shown in Figure 11.20 to be able to use the standard approach. Details of such procedures can be found in Whitaker *et al.* (1989) and Cabello *et al.* (1994). The value ϕ_B at dummy node B might be calculated by averaging over nearby nodal values. Thus, if ϕ_B was available

$$r = \frac{\phi_P - \phi_B}{\phi_A - \phi_P} \tag{11.39}$$

Figure 11.20 Upwind dummy node reconstruction for higher-order schemes

In the absence of ϕ_B, Darwish and Moukalled (2003) recommend

$$r = \left[\frac{(2\nabla\phi_P \,.\, \mathbf{r}_{PA})}{\phi_A - \phi_P} - 1 \right] \tag{11.40}$$

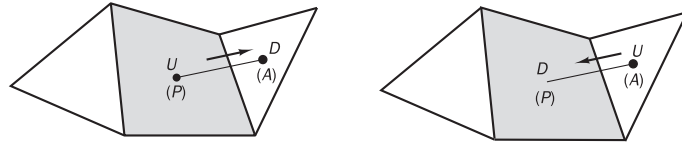
Here $\mathbf{r}_{PA}$ is the distance vector between nodes P and A. The flow can be from P to A or from A to P. To generalise the above expression we should adopt the notation 'U' for upstream and 'D' for downstream:

$$\boxed{r = \left[\frac{(2\nabla\phi_P \,.\, \mathbf{r}_{PA})}{\phi_D - \phi_U} - 1 \right]} \tag{11.41}$$

The **TVD expression for convective flux** can also be written as

$$\boxed{\phi_i = \phi_U + \frac{\psi(r)}{2}(\phi_D - \phi_U)} \tag{11.42}$$

where U denotes the upstream node and D denotes the downstream node. Depending on the direction of the flow vector along the line joining centroids of the cells, the upstream and downstream points have to be selected appropriately and allocated to P and A: see Figure 11.21. The interested reader should consult Darwish and Moukalled (2003) for further details.

Figure 11.21 Selection of upstream and downstream nodes depending on flow direction

11.10 Treatment of source terms

Finally the source term in equation (11.4) is treated in the same way as we did in Cartesian coordinates:

$$\int_{CV} S\mathrm{d}V = \bar{S}\Delta V \tag{11.43}$$

where ΔV is the volume of the control volume
$\bar{S}$ is the average of S over the control volume

A second-order accurate approximation of integral (11.43) is obtained using the midpoint rule, which replaces the average $\bar{S}$ by the nodal value of the source function S evaluated at the centroid of the control volume. The source term is introduced to the discretised equation as before by using $\bar{S}\Delta V = S_u - S_p\phi_P$. In 2D the volume is the area of the cell multiplied by unit dimension in the direction normal to the 2D plane. In 3D ΔV is the volume of the control volume and can be calculated using standard geometrical relationships and vector algebra. Kordula and Vinokur (1983), for example, give a method to calculate volumes in an efficient manner.

11.11 Assembly of discretised equations

The diffusion flux through a face is

$$D_i(\phi_A - \phi_P) + S_{D\text{-}cross,i} \tag{11.44}$$

Using a TVD scheme for convective flux and treating the TVD contribution as deferred correction as outlined in Chapter 5, the convective flux through a face is

$$F_i[\phi_U + \psi(r)(\phi_D - \phi_U)/2] \tag{11.45}$$

The source term for the volume is

$$S_u + S_p\phi_P \tag{11.46}$$

When these are substituted into steady flow equation (11.4)

$$\int_A n\,.\,(\rho\phi\mathbf{u})\,\mathrm{d}A = \int_A n\,.\,(\Gamma\ \mathrm{grad}\ \phi)\,\mathrm{d}A + \int_{CV} S_\phi\,\mathrm{d}V \tag{11.47}$$

we obtain

$$\sum_{\text{all surfaces}} F_i[\phi_U + \psi(r)(\phi_D - \phi_U)/2] = \sum_{\text{all surfaces}} [D_i(\phi_A - \phi_P) + S_{D\text{-}cross,i}] + (S_u + S_p\phi_P) \tag{11.48}$$

In the above equation A stands for the centroid of each control volume surrounding the point P. For the convective terms U and D have to be appropriately allocated to P and A depending on the flow direction across the face. The use of vector algebra in the derivation of the relevant equations, in conjunction with the definitions of the unit normal vectors and velocity vectors, takes care of the flow direction. We automatically recover the correct magnitude *and* sign of F_i.

The above equation can be rearranged as

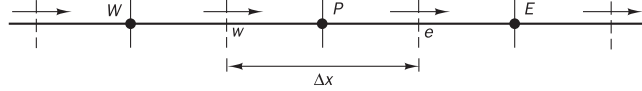
$$a_P\phi_P = \sum a_{nb}\phi_{nb} + S_u + S_u^{DC} + \sum S_{D\text{-}cross,i} \tag{11.49}$$

where $a_P = \sum a_{nb} - S_P + \sum F_i$

Here S_u^{DC} is a source term arising from deferred corrections from TVD or higher-order schemes (see section 5.10). $\sum S_{D\text{-}cross,i}$ is the source term due to cross-diffusion and $\sum F_i$ is the mass imbalance over all faces. Note that the system of equations arising from the discretisation process is no longer a banded matrix, since, depending on the shape of the control volume, the nodes for transported quantity ϕ may be connected to an arbitrary number of neighbouring nodes in an unstructured mesh. Solution of the system therefore requires techniques such as the multigrid method described in Chapter 7 or the conjugate gradient method.

Application of the unstructured equations to Cartesian grids

We solve a source-free 1D convection–diffusion problem shown in Figure 11.22 using upwind differencing to demonstrate that we can recover the Cartesian discretised equations presented in section 5.6 from equation (11.48).

Figure 11.22 A 1D fluid flow problem

The essential parameters used in equation (11.48) are control volume width $= \Delta x$, the distance between nodes $\Delta\xi = \Delta x$. For equally spaced control volumes the distance between nodes is the same, i.e. $\Delta x_{PE} = \Delta x_{WP} = \Delta x$. The outward normal vector for the east face is

$$\mathbf{n}_e = 1\mathbf{i} + 0\mathbf{j}$$

The vector $\mathbf{e}_\xi$ for the line PE is

$$\mathbf{e}_{PE} = 1\mathbf{i} + 0\mathbf{j}$$

and the area of the east face is

$$\Delta A_e = 1.0$$

The outward normal vector for the west face is

$$\mathbf{n}_w = -1\mathbf{i} + 0\mathbf{j}$$

The vector $\mathbf{e}_\xi$ for the line PW is

$$\mathbf{e}_{PW} = -1\mathbf{i} + 0\mathbf{j}$$

The area of the west face is

$$\Delta A_w = 1.0$$

and the convection velocity vector is

$$\mathbf{u} = u\mathbf{i} + 0\mathbf{j}$$

We use the standard notation for fluid properties adopted in the previous chapters: the diffusion coefficient is denoted by Γ and the density by ρ.

Since the faces of the control volumes are perpendicular to the lines joining nodes, no cross-diffusion terms arise in this orthogonal grid. Hence, $S_{D\text{-}cross} = 0$ and there is no source term. Thus, the diffusion flux is given by (11.22):

$$\mathbf{n}\,.\,\Gamma\ \text{grad}\ \phi\Delta A_i = D_i(\phi_A - \phi_P)$$

$$\text{where } D_i = \frac{\Gamma}{\Delta x}\frac{\mathbf{n}\,.\,\mathbf{n}}{\mathbf{n}\,.\,\mathbf{e}_\xi}\Delta A_i$$

Diffusion flux parameters D_i for the west and east faces from equation (11.24) are

$$D_e = \frac{\Gamma}{\Delta x}\frac{(1\mathbf{i} + 0\mathbf{j})\,.\,(1\mathbf{i} + 0\mathbf{j})}{(1\mathbf{i} + 0\mathbf{j})\,.\,(1\mathbf{i} + 0\mathbf{j})}1.0 = \frac{\Gamma}{\Delta x} = D$$

$$D_w = \frac{\Gamma}{\Delta x}\frac{(-1\mathbf{i} + 1\mathbf{j})\,.\,(-1\mathbf{i} + 1\mathbf{j})}{(-1\mathbf{i} + 1\mathbf{j})\,.\,(-1\mathbf{i} + 1\mathbf{j})}1.0 = \frac{\Gamma}{\Delta x} = D$$

The mass flow rate through the east face is

$$F_e = \rho(1\mathbf{i} + 0\mathbf{j})\,.\,(u\mathbf{i} + 0\mathbf{j})1.0 = \rho u = F$$

and the mass flow rate through the west face is

$$F_w = \rho(-1\mathbf{i} + 0\mathbf{j}) \, . \, (u\mathbf{i} + 0\mathbf{j})1.0 = -\rho u = -F$$

Let us use the upwind scheme; then $\psi(r) = 0$ in equation (11.48) and

$$\sum_{\text{all surfaces}} F_i[\phi_U + \psi(r)(\phi_U - \phi_D)/2] = \sum_{\text{all surfaces}} D_i(\phi_A - \phi_P) + (S_u + S_p\phi_P) \quad (11.50)$$

Now if we apply equation (11.48) with the parameters calculated above we obtain

$$[F_e(\phi_P + 0) + F_w(\phi_W + 0)] = [D_e(\phi_E - \phi_P)] + [D_w(\phi_W - \phi_P)] + (S_u + S_p\phi_P) \quad (11.51)$$

$$F_e\phi_P + F_w\phi_W = D_e\phi_E - D_e\phi_P + D_w\phi_W - D_w\phi_P + (S_u + S_p\phi_P) \quad (11.52)$$

Equation (11.52) can be rearranged in the form

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + S_u \quad (11.53)$$

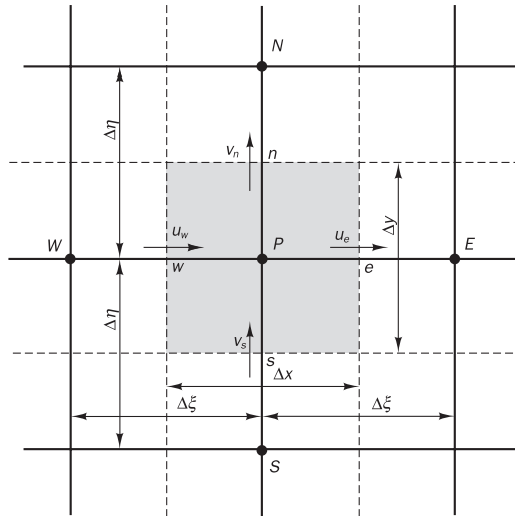
where $a_W = D_w - F_w \quad a_E = D_e \quad a_P = a_W + a_E - S_p + (F_e + F_w)$

At first sight the above expressions for the coefficients appear to be slightly different from those in (5.30). However, since $F_w = -F$ and $F_e = F$, we obtain discretised equation (5.31). Thus, we have

$$a_W = D + F \quad a_E = D \quad a_P = a_W + a_W - S_p + (F - F)$$

It is important to note that F_w has a magnitude *and* a sign in these unstructured grid calculations. In fact, it is negative $(-F)$ in this example, so equation (11.52) is the same as equation (5.29) in Chapter 5, where F_w was treated as a magnitude, i.e. an unsigned quantity. The use of vector algebra in the derivation of the relevant equations allows us to recover the correct magnitude and sign of F_w.

We show that Cartesian 2D expressions can also be recovered with relative ease. Consider the 2D situation shown in Figure 11.23.

Figure 11.23 A 2D Cartesian grid arrangement

The essential parameters used in equation (11.48) are:

Control volume width in x-direction: $\Delta\xi = \Delta x$ for the lines PE and WP
Control volume width in y-direction: $\Delta\eta = \Delta y$ for the lines PN and SP

For equally spaced control volumes the distance between nodes is the same, i.e. $\Delta x_{PE} = \Delta x_{WP} = \Delta x$ in the x-direction and $\Delta y_{PN} = \Delta y_{SP} = \Delta y$ in the y-direction. Relevant unit vectors and the area for each side of a cell are summarised in Table 11.1.

Table 11.1

Cell face	*Outward normal vector to cell face*	*Vector* $\mathbf{e}_\xi$ *for the line between P and node*	*Area of cell face*
East (e)	$\mathbf{n}_e = 1\mathbf{i} + 0\mathbf{j}$	Line PE: $\mathbf{e}_{PE} = 1\mathbf{i} + 0\mathbf{j}$	$\Delta A_e = \Delta y$
West (w)	$\mathbf{n}_w = -1\mathbf{i} + 0\mathbf{j}$	Line PW: $\mathbf{e}_{PW} = -1\mathbf{i} + 0\mathbf{j}$	$\Delta A_w = \Delta y$
North (n)	$\mathbf{n}_n = 0\mathbf{i} + 1\mathbf{j}$	Line PN: $\mathbf{e}_{PN} = 0\mathbf{i} + 1\mathbf{j}$	$\Delta A_n = \Delta x$
South (s)	$\mathbf{n}_s = 0\mathbf{i} - 1\mathbf{j}$	Line PS: $\mathbf{e}_{PS} = 0\mathbf{i} - 1\mathbf{j}$	$\Delta A_s = \Delta x$

The convection velocity vector is the same at all faces: $\mathbf{u} = u\mathbf{i} + v\mathbf{j}$, where u and v are both positive everywhere.

The other notation is just as standard: the diffusion coefficient is denoted by Γ and the density by ρ. The normal vectors of the east, west, north and south control surfaces of the control volume coincide with the lines connecting the nodes straddling these faces, so again the cross-diffusion term is zero in this orthogonal grid. The values of the diffusion flux parameter for the east, west, north and south faces from equation (11.24) are shown in Table 11.2.

Table 11.2

Diffusion flux parameter D_i *for each face*	
East face	*West face*
$D_e = \dfrac{\Gamma}{\Delta x}\dfrac{(1\mathbf{i} + 0\mathbf{j})\ .\ (1\mathbf{i} + 0\mathbf{j})}{(1\mathbf{i} + 0\mathbf{j})\ .\ (1\mathbf{i} + 10\mathbf{j})}\Delta y\ .\ 1.0 = \dfrac{\Gamma}{\Delta x}\Delta y$	$D_w = \dfrac{\Gamma}{\Delta x}\dfrac{(-1\mathbf{i} + 0\mathbf{j})\ .\ (-1\mathbf{i} + 0\mathbf{j})}{(-1\mathbf{i} + 0\mathbf{j})\ .\ (-1\mathbf{i} + 0\mathbf{j})}\Delta y = \dfrac{\Gamma}{\Delta x}\Delta y$
North face	*South face*
$D_n = \dfrac{\Gamma}{\Delta y}\dfrac{(0\mathbf{i} + 1\mathbf{j})\ .\ (0\mathbf{i} + 1\mathbf{j})}{(0\mathbf{i} + 1\mathbf{j})\ .\ (0\mathbf{i} + 1\mathbf{j})}\Delta x = \dfrac{\Gamma}{\Delta y}\Delta x$	$D_s = \dfrac{\Gamma}{\Delta y}\dfrac{(0\mathbf{i} - 1\mathbf{j})\ .\ (0\mathbf{i} - 1\mathbf{j})}{(0\mathbf{i} - 1\mathbf{j})\ .\ (0\mathbf{i} - 1\mathbf{j})}\Delta x = \dfrac{\Gamma}{\Delta y}\Delta x$

Table 11.3 shows the mass flow rates through each face, which are calculated using $F_i = \mathbf{n}\ .\ (\rho\mathbf{u})\Delta A_i$.

Table 11.3

Mass flow rate F_i through each face	
East face	***West face***
$F_e = \rho(1\mathbf{i} + 0\mathbf{j}) \,.\, (u\,\mathbf{i} + v\,\mathbf{j})\Delta y = \rho u \Delta y$	$F_w = \rho(-1\mathbf{i} + 0\mathbf{j}) \,.\, (u\,\mathbf{i} + v\,\mathbf{j})\Delta y = -\rho u \Delta y$
North face	***South face***
$F_n = \rho(0\mathbf{i} + 1\mathbf{j}) \,.\, (u\,\mathbf{i} + v\,\mathbf{j})\Delta x = \rho v \Delta x$	$F_s = \rho(0\mathbf{i} - 1\mathbf{j}) \,.\, (u\,\mathbf{i} + v\,\mathbf{j})\Delta x = -\rho v \Delta x$

As in the 1D example we use the upwind scheme, $\psi(r) = 0$.
Now we apply equation (11.48):

$$\sum_{\text{all surfaces}} F_i[\phi_U + \psi(r)(\phi_U - \phi_D)/2] = \sum_{\text{all surfaces}} [D_i(\phi_A - \phi_P) + S_{D\text{-}cross,i}] + (S_u + S_p\phi_P) \qquad (11.54)$$

If we substitute the information we have generated above, we obtain

$$\begin{aligned}[F_e(\phi_P + 0) + F_w(\phi_W + 0) + F_n(\phi_P + 0) + F_s(\phi_S + 0)] \\ = [D_e(\phi_E - \phi_P) + 0] + [D_w(\phi_W - \phi_P) + 0] \\ + [D_n(\phi_N - \phi_P) + 0] + [D_s(\phi_S - \phi_P) + 0] + (S_u + S_p\phi_P)\end{aligned} \qquad (11.55)$$

$$\begin{aligned}F_e\phi_P + F_w\phi_W + F_n\phi_P + F_s\phi_S = D_e\phi_E - D_e\phi_P + D_w\phi_W - D_w\phi_P + D_n\phi_N - D_n\phi_P \\ + D_s\phi_S - D_s\phi_P + (S_u + S_p\phi_P)\end{aligned} \qquad (11.56)$$

This can be rearranged in the form

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_N\phi_N + a_S\phi_S + S_u \qquad (11.57)$$

where $a_W = D_w - F_w \quad a_E = D_e \quad a_S = D_s - F_s \quad a_N = D_n$
and $a_P = a_W + a_E + a_S + a_N - S_p + (F_e + F_w + F_n + F_s)$

Noting that values of F_w and F_s have negative signs, this equation gives the same result as equation (5.31) extended to 2D. Instead of developing the discretised equation in the above manner we could have used the standard expressions for the upwind scheme. With the correct sign for the mass flow rate the upwind expression for any coefficient may be written as

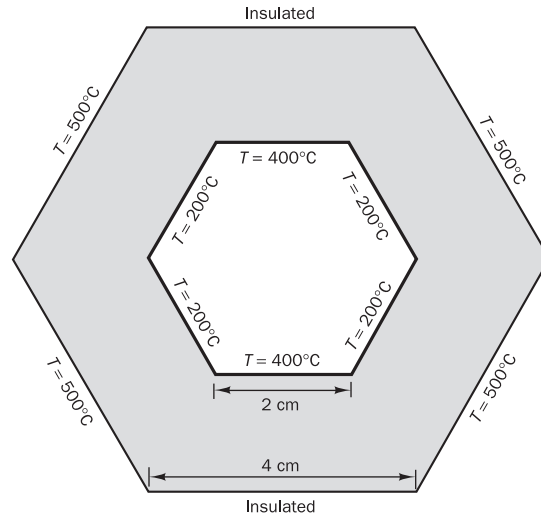
$$a_i = D_i + \max(-F_i, 0) \qquad (11.58)$$

It can be easily seen that this also results in the correct values for the coefficients in the above example.

11.12 Example calculations with unstructured grids

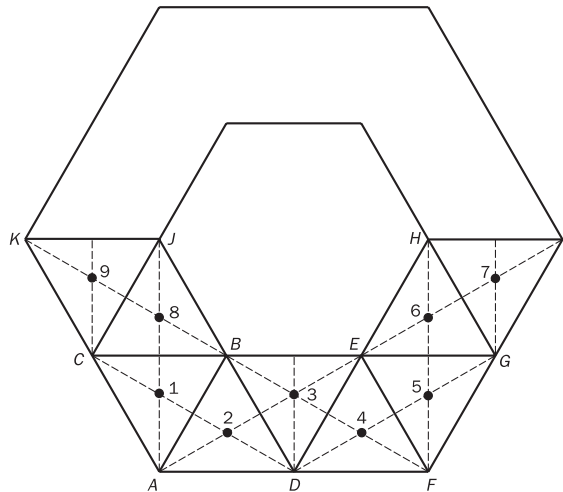
Example 11.1

Consider the 2D hexagonal ring geometry shown in Figure 11.24. It is required to calculate the temperature distribution given the temperature and flux boundary conditions in the figure. The thermal conductivity of the material is $k = 50$ W/m.K.

Figure 11.24 The geometry and boundary conditions

Solution

The problem involves conduction only, so it is a diffusion problem without sources. The geometry does not fit into Cartesian or cylindrical co-ordinates, therefore an unstructured mesh is required. We use triangular cells, which are an obvious choice for this problem. Alternatively, a mesh constructed of quadrilateral cells is also possible.

To demonstrate the method we have chosen a triangular mesh with equilateral triangles, which has the advantage that it is an orthogonal unstructured grid, as shown in Figure 11.25. Since the normal vector to any face lies exactly along the lines joining centroids, we do not need to calculate the cross-diffusion term arising from non-orthogonality. Due to the symmetry of the problem domain and the boundary conditions only a quarter of the problem is required in the calculation. However, we would not be able to use equilateral triangles everywhere, so it is more convenient to consider one half of the geometry and use symmetry boundary conditions across lines *HI* and *KJ*.

Figure 11.25 The grid used and the notation

The governing equation of heat conduction is

$$\mathrm{div}(k \ \mathrm{grad} \ T) = 0 \tag{11.59}$$

Notations based on west, east, north and south nodes are meaningless in unstructured grids and it is easier to refer to nodes by numbers. The coefficients associated with each node will also be referred to by numbers. However, we still use P to identify the central node under consideration.

We can use (11.48) directly to write the discretised equation for each triangular control volume as follows:

$$\sum_{\text{all surfaces}} D_i(T_{nb} - T_P) = 0 \tag{11.60}$$

where nb is the node number of the adjacent cell.

As illustrated earlier, the final discretised equation has the following form:

$$a_P T_P = \Sigma a_{nb} T_{nb} \tag{11.61}$$

$$\text{where } a_P = \Sigma a_{nb} - S_P$$

The adopted mesh is so simple that many geometrical quantities required in the calculation can be easily deduced using simple trigonometry. In a general situation these would be calculated using vector algebra.

The area of all control volume faces is

$$\Delta A_i = \Delta\eta = 2 \times 10^{-2} \ \mathrm{m}^2$$

and the distance between nodes is

$$\Delta\xi = 2/\sqrt{3} \times 10^{-2} \ \mathrm{m}$$

As the mesh is orthogonal the value of $(\mathbf{n} \ . \ \mathbf{n}/\mathbf{n} \ . \ \mathbf{e}) = 1$ for this case.

The given boundary temperatures are as follows: $T_{AC} = 500°\mathrm{C}$, $T_{CK} = 500°\mathrm{C}$, $T_{BE} = 400°\mathrm{C}$, $T_{EH} = 200°\mathrm{C}$, $T_{BJ} = 200°\mathrm{C}$, $T_{FG} = 500°\mathrm{C}$ and $T_{GI} = 500°\mathrm{C}$. Edges AD and DF are insulated (zero-flux) boundaries.

Node 1

Flux through any face is

$$D_i(T_N - T_P) = \frac{k}{\Delta\xi}\Delta A_i(T_N - T_P)$$

Flux through face AB is

$$k \times \frac{(T_2 - T_P)}{2/\sqrt{3} \times 10^{-2}} \times 2 \times 10^{-2} = k\sqrt{3}\,(T_2 - T_P)$$

Flux through face BC is

$$k \times \frac{(T_8 - T_P)}{2/\sqrt{3} \times 10^{-2}} \times 2 \times 10^{-2} = k\sqrt{3}\,(T_8 - T_P)$$

Face AC is a boundary, so the flux through this face is introduced as a source term using the unstructured mesh equivalent of the half-cell approximation first introduced in section 4.3:

Flux through AC is

$$k\frac{(T_{AC} - T_P)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2} = 2k\sqrt{3}(T_{AC} - T_P)$$

Summation of fluxes through all three faces gives

$$k\sqrt{3}(T_2 - T_P) + k\sqrt{3}\,(T_8 - T_P) + 2k\sqrt{3}\,(T_{AC} - T_P) = 0$$

This can be simplified to

$$(1 + 1 + 2)T_P = T_2 + T_8 + 2 \times T_{AC}$$

$$4T_1 = T_2 + T_8 + 1000 \quad (11.62)$$

It is not necessary to go through this derivation for every cell; we may calculate coefficients using standard expressions for the coefficients of the discretised equation and introduce boundary conditions as source terms. In this example, coefficients connecting any neighbour node are given by $a_i = D_i = (k/\Delta\xi)\Delta A_i$, where ΔA_i is the area of the face and $\Delta\xi$ is the distance between the nodes. In our mesh the value of all coefficients is equal to

$$\frac{k}{(2/\sqrt{3}) \times 10^{-2}} 2 \times 10^{-2} = k\sqrt{3}$$

Node 2
$a_1 = k\sqrt{3}$, $a_3 = k\sqrt{3}$, and the face AD is an insulated boundary.
Flux through the boundary AD is zero (insulated boundary), so

$$S_u = 0$$
$$S_P = 0$$
$$a_P = a_1 + a_3 - S_P$$

After simplification the discretised equation for node 2 is

$$2T_2 = T_1 + T_3 \quad (11.63)$$

Node 3
BE is a constant temperature boundary, $T_{BE} = 400°\text{C}$, and $a_2 = k\sqrt{3}$, $a_4 = k\sqrt{3}$.
Flux through BE is

$$k\frac{(T_{BE} - T_P)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2} = 2k\sqrt{3}(T_{BE} - T_P)$$

$$S_u = 2k\sqrt{3}T_{BE}$$

$$S_P = -2k\sqrt{3}$$

$$a_P = a_4 + a_6 - S_P = (1 + 1 + 2)k\sqrt{3}$$

The discretised equation for node 3 is

$$4T_3 = T_2 + T_4 + 800 \quad (11.64)$$

Node 4
Same as node 2; boundary DF is an insulated boundary (no flux).
The discretised equation for node 4 is

$$2T_4 = T_5 + T_3 \quad (11.65)$$

Node 5
Same as node 1; boundary FG is a constant temperature boundary.
Flux through FG is

$$k\frac{(T_{FG}-T_P)}{(1/\sqrt{3})\times 10^{-2}}\times 2\times 10^{-2}=2k\sqrt{3}(T_{FG}-T_P)$$
$$S_u=2k\sqrt{3}T_{FG}$$
$$S_P=-2k\sqrt{3}$$
$$a_P=a_4+a_6-S_P=(1+1+2)k\sqrt{3}$$

The discretised equation for node 5 is

$$4T_5=T_4+T_6+1000 \tag{11.66}$$

Nodes 6 and 8
Similar to node 3; face EH is a constant temperature boundary, $T_{EH}=200°C$, and face BJ is a constant temperature boundary, $T_{BJ}=200°C$.
The discretised equation for node 6 is

$$4T_6=T_5+T_7+400 \tag{11.67}$$

The discretised equation for node 8 is

$$4T_8=T_1+T_9+400 \tag{11.68}$$

Node 7
Face HI is a symmetry boundary, so no flux. Face GI is a constant temperature boundary, $T_{GI}=500°C$.
Flux through GI is

$$k\frac{(T_{GI}-T_P)}{(1/\sqrt{3})\times 10^{-2}}\times 2\times 10^{-2}=2k\sqrt{3}(T_{GI}-T_P)$$
$$S_u=2k\sqrt{3}T_{GI}$$
$$S_P=-2k\sqrt{3}$$
$$a_P=a_6-S_P=(1+2)k\sqrt{3}$$

The discretised equation for node 7 is

$$3T_7=T_6+1000 \tag{11.69}$$

Node 9
Same as node 7.
The discretised equation for node 9 is

$$3T_9=T_8+1000 \tag{11.70}$$

Summarising the discretised equations:

$$4T_1=T_2+T_8+1000$$
$$2T_2=T_1+T_3$$
$$4T_3=T_2+T_4+800$$
$$2T_4=T_5+T_3$$
$$4T_5=T_4+T_6+1000$$
$$4T_6=T_5+T_7+400$$
$$3T_7=T_6+1000$$
$$4T_8=T_1+T_9+400$$
$$3T_9=T_8+1000$$

Next, we rewrite this set of equations in matrix form:

$$\begin{bmatrix} 4 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 4 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 3 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ T_7 \\ T_8 \\ T_9 \end{bmatrix} = \begin{bmatrix} 1000 \\ 0 \\ 800 \\ 0 \\ 1000 \\ 400 \\ 1000 \\ 400 \\ 1000 \end{bmatrix} \tag{11.71}$$

We note that the coefficient matrix is not banded. Due to the chosen control volume numbering the off-diagonal entries appear at (1, 8) and (8, 1) in this matrix. The solution is

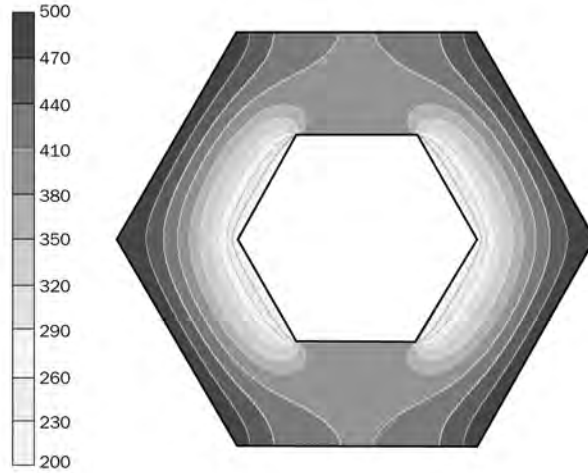
$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \\ T_6 \\ T_7 \\ T_8 \\ T_9 \end{bmatrix} = \begin{bmatrix} 435.6436 \\ 423.7624 \\ 411.8812 \\ 423.7624 \\ 435.6436 \\ 318.8119 \\ 439.6040 \\ 318.8119 \\ 439.6040 \end{bmatrix} \tag{11.72}$$

We do not have an analytical solution for comparison of our numerical results. However, we can make several checks. First it can be seen that the solution obtained is symmetric, i.e. $T_2 = T_4$, $T_1 = T_5$, $T_6 = T_8$ and $T_7 = T_9$. We can also check the flux balances across conducting boundary surfaces (Table 11.4).

Table 11.4 Heat flux at conducting boundaries

Face	*Heat flux is given by*	*Numerical values*	*Flux value*
AC	$k\dfrac{(T_{AC} - T_1)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	$50\dfrac{(500 - 435.6436)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	11146.8555
BE	$k\dfrac{(T_{BE} - T_3)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	$50\dfrac{(400 - 411.8812)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	–2057.8842
FG	Same as *AC*		11146.8555
GI	$k\dfrac{(T_{GI} - T_7)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	$50\dfrac{(500 - 439.6040)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	10460.8941
EH	$k\dfrac{(T_{EH} - T_6)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	$50\dfrac{(200 - 318.8119)}{(1/\sqrt{3}) \times 10^{-2}} \times 2 \times 10^{-2}$	–20578.8247
BJ	Same as *EH*		–20578.8247
CK	Same as *GI*		10460.8941
		Net flux (heat balance)	Σ fluxes = –0.0344 ≈ 0

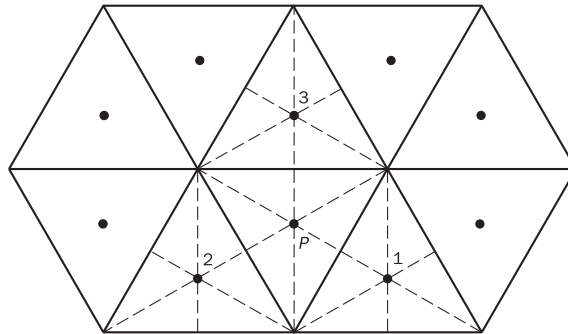
As expected, in the finite volume method the fluxes of heat into and out of the domain exactly balance. However, due to the coarseness of the grid, the temperature distribution may not be very accurate. We need to perform a high-resolution calculation to get better results. A high-resolution solution for this problem is shown in Figure 11.26.

Figure 11.26 Temperature distribution from a high-resolution solution using a fine mesh for the same problem

Convection–diffusion in a similar mesh

The method can be easily extended to convection–diffusion problems employing a similar equilateral triangular mesh. For example, with the upwind differencing method, using (11.58) the neighbour coefficients for convection–diffusion in the equilateral triangular mesh shown in Figure 11.27 can be written as

$$\begin{aligned} a_1 &= D_1 + \max(-F_1, 0) \\ a_2 &= D_2 + \max(-F_2, 0) \\ a_3 &= D_3 + \max(-F_3, 0) \end{aligned} \tag{11.73}$$

$$\text{where } D_i = \frac{k}{\Delta\xi}\Delta A_i \quad \text{and} \quad F_i = \rho(\mathbf{n}\,.\,\mathbf{u})\Delta A_i$$

Figure 11.27 Part of an equilateral triangular grid for 2D calculations

The resulting discretised equation is

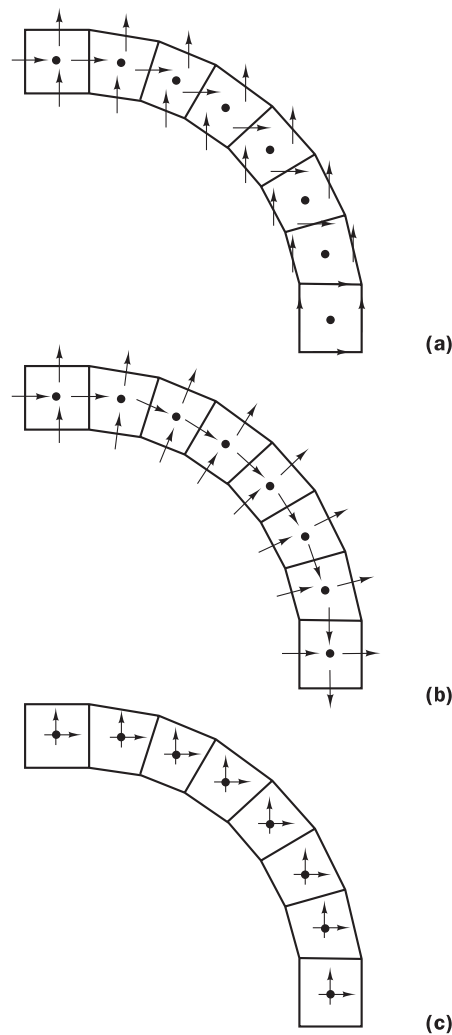
$$a_P\phi_P = \sum a_{nb}\phi_{nb} + S_u \tag{11.74}$$

where $a_P = \sum a_{nb} - S_P + (F_1 + F_2 + F_3)$
and F_1, F_2, F_3 have appropriate sign + if outwards and – if inwards

11.13 Pressure–velocity coupling in unstructured meshes

It was mentioned earlier that in unstructured and curvilinear structured grid arrangements it is most convenient to store Cartesian velocity components at the cell centres, where the pressure values are also stored. This is known as a co-located arrangement. There are other methods which use staggered arrangements, covariant and contravariant velocity components.

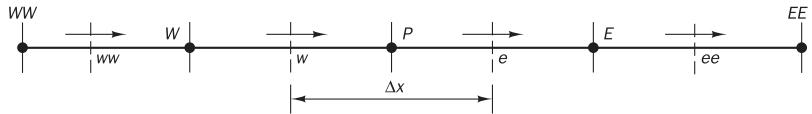
Figure 11.28 illustrates some of these methods for a sequence of cells arranged along a 90° bend. In a staggered arrangement with Cartesian

Figure 11.28 Different velocity vector arrangements for curvilinear grids: (a) staggered velocity arrangement; (b) contravariant velocity arrangement; (c) co-located velocity arrangement

velocity components (Figure 11.28a) the calculated u- and v-velocity components do not represent fluxes normal to the surface due to the gradual direction change except in the first and last elements. This can be overcome by using contravariant, grid-oriented velocity components as shown in Figure 11.28b, but the governing equations and derivation of the discretised equations are much more complex. Details of some of these special procedures can be found in Hirt *et al.* (1974), Rhie and Chow (1983), Peric (1985), Reggio and Camarero (1986) and Rodi *et al.* (1989), among others.

Here we concentrate on the co-located velocity arrangement used in conjunction with Cartesian velocity components (Figure 11.28c). In Chapter 6 we discussed the 'checker-board' pressure field effect and the reason for using a staggered grid. Therefore, when a co-located arrangement is used for pressure and velocity values we need to ensure correct pressure–velocity coupling. In the next section we explain some details of the special measures that are used in a 1D Cartesian grid with a co-located arrangement and their extension to unstructured grids. We limit our discussion to the Rhie and Chow (1983) interpolation method, which is the most common practice in co-located unstructured grids and also in curvilinear structured meshes.

11.14 Staggered vs. co-located grid arrangements

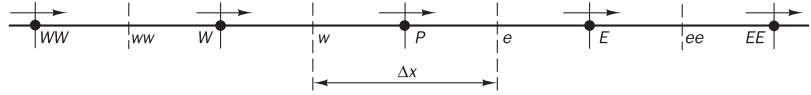
Before we explain the handling of pressure–velocity coupling in co-located arrangements we briefly revisit the staggered arrangement for a 1D grid (see Figure 11.29).

Figure 11.29 A 1D staggered grid arrangement

In this arrangement the u-momentum equation is solved at location e, and using standard notation the discretised form at velocity node e is

$$a_{P_e}u_e = \sum a_{nb}u_{nb} + (p_P - p_E)A_e + b \tag{11.75}$$

The pressure difference that contributes to the momentum equation is $(p_P - p_E)$, which straddles the node e. The velocity u_e lies on the face of the scalar control volume centred at P. The continuity equation is applied to this scalar control volume to derive the pressure correction equation of the SIMPLE algorithm. As we saw in section 6.2, the main advantage of a staggered arrangement is its capability to deal with a 'checker-board' pressure field. Although this works very well for Cartesian and cylindrical mesh systems, the arrangement is not entirely convenient for unstructured and body-fitted mesh systems. In addition to the disadvantages mentioned in section 11.13 the storage of geometric data related to four different sets of variables in 3D flows (scalars, u-, v- and w-velocity components) is inefficient. This can be avoided by storing all variables at one single location, which is the co-located arrangement.

Now consider the 1D co-located arrangement shown in Figure 11.30, where velocity u is stored and calculated at the location P.

Figure 11.30 A 1D co-located grid arrangement

The discretised u-momentum equation at node P is

$$a_{P_P}u_P = \sum a_{nb}u_{nb} + (p_w - p_e)A_P + b_P \tag{11.76}$$

Pressures are also stored at node locations W, P, E etc., so now we need to interpolate to find the required face pressures p_w and p_e. If we use linear interpolation, which is second-order accurate (i.e. truncation error $\propto \Delta x^2$) on this uniform grid, we obtain

$$(p_e - p_w) = \frac{(p_E + p_P)}{2} - \frac{(p_P + p_W)}{2} = \frac{(p_W - p_E)}{2} \tag{11.77}$$

The discretised equation (11.76) now becomes

$$a_{P_P}u_P = \sum a_{nb}u_{nb} + \frac{1}{2}(p_W - p_E)A_P + b_P \tag{11.78}$$

$$u_P = \frac{\sum a_{nb}u_{nb} + b_P}{a_{P_P}} + \frac{1}{2}\frac{A_P}{a_{P_P}}(p_W - p_E) \tag{11.79a}$$

$$u_P = \frac{\sum a_{nb}u_{nb} + b_P}{a_{P_P}} + \frac{d_P}{2}(p_W - p_E) \tag{11.79b}$$

where $d_P = A_P / a_{P_P}$

It can be seen that the momentum equation at point P does not involve the pressure at the point P, but uses nodal pressures from adjacent control volumes.

Similarly, the discretised u-momentum equation at node E is

$$a_{P_E}u_E = \sum a_{nb}u_{nb} + (p_{ee} - p_e)A_E + b_E \tag{11.80}$$

$$a_{P_E}u_E = \sum a_{nb}u_{nb} + \frac{1}{2}(p_P - p_{EE})A_E + b_E \tag{11.81}$$

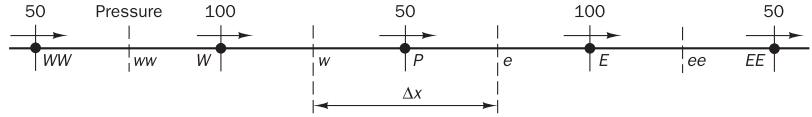
$$u_E = \frac{\sum a_{nb}u_{nb} + b_E}{a_{P_E}} + \frac{1}{2}\frac{A_E}{a_{P_E}}(p_P - p_{EE}) \tag{11.82a}$$

$$u_E = \frac{\sum a_{nb}u_{nb} + b_E}{a_{P_E}} + \frac{d_E}{2}(p_P - p_{EE}) \tag{11.82b}$$

where $d_P = A_E / a_{P_E}$

We now consider a varying pressure field as shown in Figure 11.31. On physical grounds, we expect that the large pressure differences between nodes P and E and between nodes W and P would drive flows into the control volume surrounding node P across the e and w faces, respectively, and act as sizeable momentum sources. However, pressure difference $(p_W - p_E)$ acting in the discretised momentum equations for u_P is zero. Similar considerations apply to the pressure gradient terms in the equations for u_W and u_E, which are also

zero. This is the 'checker-board' problem described earlier in Chapter 6, which is responsible for non-physical solutions as a consequence of the linear interpolation for pressure differences in a co-located arrangement.

Figure 11.31 A highly varying pressure field

Velocities on the faces *e* and *w* are used (in 1D) to discretise the continuity equation, which then gives the pressure correction equation. If the driving pressure is not properly represented in the face velocities, we cannot expect the pressure correction equation to yield a correct solution. For example, the cell face velocity u_e in the co-located arrangement has to be interpolated using u_P and u_E from equations (11.79) and (11.82). As it is, the interpolated expression for u_e will not contain a pressure gradient across the faces of the control volume and the resulting solution can be oscillatory. Rhie and Chow (1983) suggest that a higher-order term should be included to prevent this effect.

Rhie and Chow's interpolation practice is to obtain the cell face velocity u_e as

$$\boxed{u_e = \frac{u_P + u_E}{2} + \frac{1}{2}(d_P + d_E)(p_P - p_E) - \frac{1}{4}d_P(p_W - p_E) - \frac{1}{4}d_E(p_P - p_{EE})} \quad (11.83)$$

The face velocity u_e is now directly linked to the driving pressure difference $(p_P - p_E)$ across the face *e*.

The first term on the right hand side of (11.83) is the average of the velocities straddling face *e*. Comparison of the sum of (11.79b) and (11.82b) with (11.83) reveals that the last two terms in (11.83) remove the troublesome effects due to the pressure differences between nodes of adjacent control volumes. This is replaced by a driving effect $\frac{1}{2}(d_P + d_E)(p_P - p_E)$ involving the difference of pressures between the nodes straddling face *e*.

It remains to be shown that the three pressure terms in (11.83) represent the addition of a higher-order (i.e. small) term. We assume constant values for *d* everywhere and write Rhie and Chow's interpolated face velocity u_e as follows:

$$\begin{aligned} u_e &= \frac{u_P + u_E}{2} + d(p_P - p_E) - \frac{1}{4}d(p_W - p_E) - \frac{1}{4}d(p_P - p_{EE}) \\ &= \frac{u_P + u_E}{2} + \frac{d}{4}[4(p_P - p_E) - (p_W - p_E) - (p_P - p_{EE})] \end{aligned} \quad (11.84)$$

Thus, the pressure term is

$$\frac{d}{4}(p_{EE} - 3p_E + 3p_P - p_W) \quad (11.85)$$

To identify the meaning of this term we consider the third derivative of pressure across face *e*:

$$\left.\frac{\partial^3 p}{\partial x^3}\right|_e = \frac{\partial}{\partial x}\left.\left(\frac{\partial^2 p}{\partial x^2}\right)\right|_e = \frac{1}{\Delta x}\left[\left.\left(\frac{\partial^2 p}{\partial x^2}\right)\right|_E - \left.\left(\frac{\partial^2 p}{\partial x^2}\right)\right|_P\right]$$

$$= \frac{1}{\Delta x}\left[\frac{(p_{EE} - 2p_E + p_P)}{\Delta x^2} - \frac{(p_E - 2p_P + p_W)}{\Delta x^2}\right]$$

$$= \frac{1}{\Delta x^3}(p_{EE} - 3p_E + 3p_P - p_W) \tag{11.86}$$

In (11.86) the approximation of the second derivative of pressure involves central differencing. Hence,

$$(p_{EE} - 3p_E + 3p_P - p_W) = \left.\frac{\partial^3 p}{\partial x^3}\right|_e \Delta x^3 \tag{11.87}$$

Comparing (11.87) with the pressure term (11.85) we deduce that

$$u_e = \frac{u_P + u_E}{2} + \left.\frac{d}{4}\frac{\partial^3 p}{\partial x^3}\right|_e \Delta x^3 \tag{11.88}$$

This shows that Rhie and Chow's pressure interpolation practice involves the addition of a third-order pressure gradient term. Since the remainder of the method is at best second-order accurate, this addition does not compromise the solution accuracy. Its beneficial effect is to provide damping of the spurious oscillations due to the co-located arrangement, so it is called a pressure smoothing term or added dissipation term. The damping is caused by the restored linkage between the pressure differences across the control volume faces and the face velocities, which appear in the continuity equation. The source term of the latter equation is the mass imbalance, which, in a constant density flow, involves differences between the cell face velocities, so the addition of a third-order pressure gradient term to each of these velocities is equivalent to adding a fourth-order pressure gradient term to their differences in the resulting pressure correction equation. The Rhie and Chow procedure has been extremely successful in co-located curvilinear body-fitted and unstructured grids.

11.15 Extension of the face velocity interpolation method to unstructured meshes

The extension of the face velocity interpolation expression (11.83) to unstructured meshes is straightforward. First, it is rewritten as follows:

$$u_e = \frac{u_P + u_E}{2} + \frac{1}{2}\left(\frac{A_P \Delta x}{a_P} + \frac{A_E \Delta x}{a_E}\right)\left(\frac{(p_P - p_E)}{\Delta x}\right)$$
$$- \frac{1}{2}\left(\frac{A_P \Delta x}{a_P}(\nabla p_P) + \frac{A_E \Delta x}{a_E}(\nabla p_E)\right) \tag{11.89}$$

The first term is the average of the nodal velocities straddling face e, the second term is the pressure difference across the face multiplied by the average of d values at P and E, and the third term involves an average of the product $d(\nabla p)$ at nodes P and E, where

$$(\nabla p)_P = \frac{P_W - P_E}{2\Delta x} \text{ and } (\nabla p)_E = \frac{p_P - p_{EE}}{2\Delta x}$$

We also note that $\Delta V = A\Delta x$ = the volume of the mesh element.

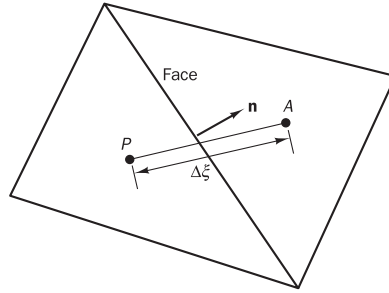
It is easy to generalise expression (11.89) to find the face velocity u_f in an unstructured mesh as shown in Figure 11.32 by using vector notation:

$$u_f = \left(\frac{\mathbf{u}_P + \mathbf{u}_A}{2}\right) . \mathbf{n} + \frac{1}{2}\left(\frac{\Delta V_P}{a_P} + \frac{\Delta V_A}{a_A}\right)\left(\frac{p_P - p_A}{\Delta \xi}\right)$$

$$- \frac{1}{2}\left[\frac{\Delta V_P}{a_P}(\nabla p)_P + \frac{\Delta V_A}{a_A}(\nabla p)_A\right] . \mathbf{e}_\xi \tag{11.90}$$

where $\Delta\xi$ is the distance between the nodes
ΔV is the volume of the control volume

The gradients $(\nabla p)_P$ and $(\nabla p)_A$ in equation (11.90) can be evaluated using an appropriate gradient reconstruction method (see section 11.9).

Figure 11.32 Control volume face of an arbitrary-shaped cell

Using the above expression (11.90) to find interpolated face velocities we can derive the SIMPLE algorithm for co-located meshes by following the steps of the standard algorithm. Velocity corrections and pressure corrections are used as in the standard algorithm for Cartesian grids:

Corrected velocity is obtained from $u = u^* + u'$ (11.91)

Corrected pressure is obtained from $p = p^* + p'$ (11.92)

For example, for the 1D situation shown in Figure 11.30, as in the standard SIMPLE algorithm the velocity correction at a face is obtained from

$$u'_e = d_e(p'_P - p'_E) \tag{11.93}$$

Using interpolated face velocities (11.83) and velocity corrections (11.93) the mass fluxes across the faces of the control volume are calculated. Discretisation of the continuity equation using these mass fluxes gives the pressure correction equation in the form

$$a_p p'_P = \sum a_{nb} p'_{nb} + b' \tag{11.94}$$

where $a_E = \rho d_e A_e$, $a_W = \rho d_w A_w$, $a_P = a_W + a_E$ and $b' = F^*_w - F^*_e$ for 1D. Similar expressions are available for 2D, 3D and for unstructured arrangements.

11.16 Summary

We have considered the different mesh arrangements that can be used to discretise the flow equations with the finite volume method. The available options can be categorised as structured and unstructured mesh arrangements. The category of **structured meshes** includes the following:

- *Cartesian meshes* involve the governing equations in their simplest form, but the discretisation of flow problems with curved domain boundaries is inaccurate.
- *Body-fitted meshes* are based on the mapping of the physical domain onto a computational domain. The approach can accommodate curved boundaries, but equations in curvilinear co-ordinates are much more complex, and there are severe difficulties in finding viable mappings for very complex geometry (pent-roof IC engine geometry).
- *Block-structured meshes* are built up out of sub-regions, each of which is meshed separately. This technique can overcome many of the difficulties associated with complex geometries and enables improved mesh quality in complex cases.

The category of **unstructured meshes** does not involve a structure of grid lines. Control volumes may have arbitrary shapes, which simplifies the meshing of complex geometries. The approach is now widely used in industrial CFD. We have outlined the following:

- Special expressions for the diffusion and convection fluxes across control volume boundaries in unstructured grids to:
  - (i) account for cross-diffusion effects due to non-orthogonality
  - (ii) cope with the absence of clear upwind co-ordinate direction to handle convection
  - (iii) avoid false diffusion by higher-order upwind or TVD convection schemes.
- Rhie and Chow's pressure interpolation for pressure–velocity coupling in a co-located arrangement for economical storage of pressure and velocity data. This special pressure–velocity coupling prevents the checker-board pressure effect, i.e. spurious oscillations in the solution.
- The SIMPLE algorithm can be used without further modification to solve flow problems.

Chapter twelve CFD modelling of combustion

12.1 Introduction

Combustion is one of the most important processes in engineering, which involves turbulent fluid flow, heat transfer, chemical reaction, radiative heat transfer and other complicated physical and chemical processes. Typical engineering applications include internal combustion engines, power station combustors, aeroengines, gas turbine combustors, boilers, furnaces, and much other combustion equipment. It is important to be able to predict the flow, temperatures, resulting species concentrations and emissions from various combustion systems for the design and improvement of combustion equipment, particularly with the current concerns about CO_2 and other emission levels and their effects on the environment. CFD lends itself very well to the modelling of combustion. Combustion processes are governed by basic transport equations for fluid flow and heat transfer with additional models for combustion chemistry, radiative heat transfer and other important sub-processes. In this chapter we attempt to outline some of the popular CFD modelling techniques used for combustion modelling. Combustion is a complex subject, and combustion modelling therefore requires a considerable amount of knowledge and experience. The material presented in this chapter is very much introductory, allowing a novice combustion modeller to gain knowledge of some basic CFD-based techniques with a view to understanding more advanced and detailed techniques in the specialist literature.

There are many types of combustion processes. Gaseous fuel combustion, liquid fuel combustion, spray combustion, solid fuel combustion, pulverised fuel combustion are a few of the many other processes used in a wide variety of application areas. To illustrate the application of CFD we concentrate on **gaseous combustion**. For other processes the reader should consult the relevant literature to find out how CFD has been successfully applied in areas like spray combustion (Beck and Watkins, 2004), pulverised coal combustion (Lockwood *et al*., 1980, 1986), diesel and spark ignition engines (Blunsdon *et al*., 1992, 1993; Henson and Malalasekera, 2000) as a modelling tool.

Gaseous combustion involves a chemical reaction between a fuel and an oxidant that are both in the gas phase. There are two categories of gaseous combustion processes: premixed combustion and non-premixed combustion. For example, combustion in a spark ignition internal combustion engine (petrol engine) can be categorised as premixed combustion, as the fuel (gasoline) is mixed with air prior to combustion, which takes place after spark ignition. Similarly the flame in the familiar Bunsen burner is also premixed combustion, as air is allowed to mix with gas prior to combustion. By contrast a jet flame where the fuel enters ambient air and is allowed to burn is an example of a non-premixed flame. The gaseous fuel mixes with the oxidant stream (air) and then combustion takes place where the conditions are right

for combustion. Non-premixed flames are also called diffusion flames because fuel and oxygen are introduced to the combustion zone in two or more separate streams and are subsequently brought together due to diffusion and mixing prior to combustion.

The bulk of this chapter is concerned with the modelling of non-premixed combustion. We do not intend discuss combustion theory in detail here, but outline some important sub-topics first: (i) thermodynamics of combustion, (ii) enthalpies of formation, (iii) transport processes, (iv) chemical kinetics, (v) equilibrium in gases, (vi) adiabatic flame temperatures. The reader should be familiar with the fundamentals of these basic concepts in order to appreciate the modelling techniques used to apply CFD to combustion calculations. The reader is referred to standard combustion texts such as Warnatz *et al.* (2001), Kuo (1986), Turns (2000), Williams (1985) etc. for thorough treatments of the fundamentals of combustion processes. Next we explain in some detail the calculation of non-premixed combustion, and conclude the chapter with a brief description of CFD modelling of premixed combustion.

12.2 Application of the first law of thermodynamics to a combustion system

We use the volume V and the absolute temperature T as the two variables to describe the thermodynamic state of a combusting fluid system. Initially, the system contains fuel and air as reactants – referred to by subscript R – at a state of (V_1, T_1). After combustion has taken place the system contains reaction products – indicated by subscript P – at a state of (V_2, T_2). We may apply the first law of thermodynamics to this system. If the system boundaries are adiabatic and the process is a non-flow process the first law of thermodynamics indicates that the heat released by the chemical reaction should be equal to the change in the internal energy U between initial state '1' and final state '2'. In order to avoid accounting problems in the internal energy balance the heat release is evaluated at a reference state:

$$U_{P2} - U_{R1} = (U_{P2} - U_{P0}) + (U_{P0} - U_{R0}) + (U_{R0} - U_{R1}) \qquad (12.1)$$

In the above equation U is the total internal energy and subscript '0' denotes the reference state. The reactants are first considered to be brought from their initial state '1' to state '0' at which the reaction takes place and then the resulting products are taken to state '2'. The term $(U_{P0} - U_{R0})$ is called the internal energy of combustion and given the notation ΔU_0.

Usually reactants and products are gaseous mixtures. In this case the internal energy changes associated with bringing reactants from state '1' to state '0' and products from state '0' to state '2' can be expressed in terms of temperature by assuming mean values for the specific heat c_v:

$$U_{R0} - U_{R1} = \sum_{i}^{\text{all reactants}} m_i c_{vi}(T_0 - T_1)$$

$$U_{P2} - U_{P0} = \sum_{i}^{\text{all products}} m_i c_{vi}(T_2 - T_0)$$

Here m_i is the mass of species i in the mixture and c_{vi} is the mean specific heat at constant volume of species i.

Similarly for open systems involving flow processes we need to consider changes of enthalpy (kinetic energy changes are small compared with the heat released during combustion processes and can often be neglected):

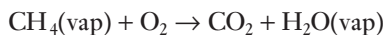
$$H_{P2} - H_{R1} = (H_{P2} - H_{P0}) + (H_{P0} - H_{R0}) + (H_{R0} - H_{R1}) \qquad (12.2)$$

where H is total enthalpy and the change of enthalpy at the reference state $(H_{P0} - H_{R0}) = \Delta H_0$ is called the enthalpy of combustion. Assuming mean c_p-values for products and reactants we have

$$H_{R0} - H_{R1} = \sum_{i}^{\text{all reactants}} m_i c_{pi} (T_0 - T_1)$$

$$H_{P2} - H_{P0} = \sum_{i}^{\text{all products}} m_i c_{pi} (T_2 - T_0)$$

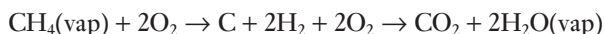
The reference state is usually taken as 25°C and 1 atm (101.3 kPa) pressure. Property values at this standard reference state are indicated by subscript '0'. For various fuels, values of Δu_0 and Δh_0 (per kmol) are available in thermodynamic property tables such as Rogers and Mayhew (1994). For example,

$$CH_4(\text{vap}) + O_2 \rightarrow CO_2 + H_2O(\text{vap})$$

From tables, heat of combustion for this reaction is $\Delta h_0 = -802310$ kJ/kmol, at 25°C.

Depending on the phase of the products (vapour or liquid) Δh_0 values can be different. The higher value is obtained when the components of products such as water are in a condensed state. Since it is not possible to list Δh_0 for all fuels and mixtures a more fundamental property known as enthalpy of formation is widely used in combustion calculations.

12.3 Enthalpy of formation

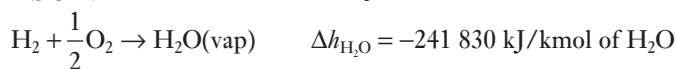
The enthalpy of formation Δh_f is defined as the increase in enthalpy when a compound is formed from its constituent elements in their natural forms at standard temperature and pressure. The elements themselves have zero enthalpy of formations in their naturally occurring states. For example, values of Δh_f for O_2, N_2, H_2 and C are zero as they are formed from their elements. However, CO_2 is formed by the reaction $C_{(graphite)} + O_2 \rightarrow CO_2$. A large amount of heat is liberated during this exothermic reaction. The enthalpy of formation is −393 520 kJ/kmol of CO_2. Enthalpy of formation values for various reactions are available in thermodynamic and combustion reference sources (Rogers and Mayhew, 1994; Kuo, 2005). To calculate the enthalpy of combustion for a fuel (sometimes called the enthalpy of reaction) we may view the reaction as a recombination of elements of the fuel. Thus, the enthalpy of reaction may be calculated using **enthalpies of formation**. For example, combustion of CH_4 may be broken down to

$$CH_4(\text{vap}) + 2O_2 \rightarrow C + 2H_2 + 2O_2 \rightarrow CO_2 + 2H_2O(\text{vap})$$

$$\Delta h_{combustion} = (\Delta h_{CO_2} + 2\Delta h_{H_2O}) - (\Delta h_{CH_4} + 2\Delta h_{O_2})$$

Individual heat of formation values in kJ/kmol are

$$C_{(graphite)} + O_2 \rightarrow CO_2 \qquad \Delta h_{CO_2} = -393\ 520 \text{ kJ/kmol of } CO_2$$

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O(\text{vap}) \qquad \Delta h_{H_2O} = -241\ 830 \text{ kJ/kmol of } H_2O$$

$$C_{(graphite)} + 2H_2 \rightarrow CH_4 \qquad \Delta h_{CH_4} = -74\ 870 \text{ kJ/kmol of } CH_4$$

For combustion of CH_4

$$\begin{aligned}\Delta h_{combustion} &= (-393\,520 - 2 \times 241\,830) - (-74\,870 + 2 \times 0)\\ &= -802\,310 \text{ kJ/kmol of } CH_4\end{aligned}$$

12.4 Some important relationships and properties of gaseous mixtures

Before and after combustion the gaseous volume will usually contain a mixture of species. It is therefore important to consider the basic properties of such mixtures.

The mole fraction of species k in a mixture is defined as

$$X_k = \frac{n_k}{n_1 + n_2 + n_3 + \ldots + n_N} = \frac{n_k}{n_{total}} \tag{12.3}$$

where n_1, n_2 etc. indicate the number of moles of each species (k) in the mixture, and n_{total} is the total number of moles in the mixture.

Partial pressure is defined as

$$p_k = \frac{n_k}{n_{total}} p = X_k p \tag{12.4}$$

where p is the total pressure (which is the sum of all partial pressures by definition). Alternatively

$$X_k = \frac{p_k}{p} \tag{12.5}$$

The mass fraction of species k in a mixture is defined as

$$Y_k = \frac{m_k}{m_1 + m_2 + m_3 + \ldots + m_N} = \frac{m_k}{m_{total}} \tag{12.6}$$

where m_1, m_2 etc. indicate the mass of each species in the mixture, and m_{total} is the total mass. By definition

$$\sum_{k}^{\text{all species}} X_k = 1 \tag{12.7}$$

$$\sum_{k}^{\text{all species}} Y_k = 1 \tag{12.8}$$

The mole fractions and mass fractions are related by

$$Y_k = X_k \frac{(MW)_k}{(MW)_{mix}} \tag{12.9}$$

$$X_k = Y_k \frac{(MW)_{mix}}{(MW)_k} \tag{12.10}$$

Here $(MW)_k$ stands for molecular weight of species 'k' and $(MW)_{mix}$ is the molecular weight of the mixture given by

$$(MW)_{mix} = \sum_{k}^{\text{all species}} X_k (MW)_k \tag{12.11}$$

Alternatively

$$\frac{1}{(MW)_{mix}} = \sum_{k}^{\text{all species}} \frac{Y_k}{(MW)_k} \tag{12.12}$$

The density of a species k is defined as

$$\rho_k = \frac{p_k}{R_u/(MW)_k T} \tag{12.13}$$

Here R_u is the universal gas constant. The density of the mixture ρ is then

$$\begin{aligned} \rho &= \sum_{k}^{\text{all species}} \rho_k \\ &= \sum_{k}^{\text{all species}} \frac{p_k}{R_u/(MW)_k T} = \frac{p}{R_u T} \sum_{k}^{\text{all species}} X_k (MW)_k \\ &= \frac{p}{R_u/(MW)_{mix} T} \end{aligned} \tag{12.14}$$

The concentration of a species k is defined as the number of kilomoles of the species per unit volume, and usually denoted by symbol C_k with units kmol/m^3:

$$C_k = \frac{n_k}{V} = \frac{X_k n_{total}}{V} \tag{12.15}$$

Using the equation of state $pV = n_{total} R_u T$ the concentration can also be written as

$$C_k = \frac{X_k n_{total}}{n_{total} R_u T/p} = X_k \frac{p}{R_u T} = \frac{p_k}{R_u T} \tag{12.16}$$

Concentration may also be related to mass fraction using the relationships between mole fraction and mass fraction:

$$C_k = \frac{Y_k (MW)_{mix}}{(MW)_k} \frac{p}{R_u T} = \frac{Y_k \rho}{(MW)_k} \tag{12.17}$$

where ρ is the density of the mixture. Alternatively

$$Y_k = \frac{C_k (MW)_k}{\rho} \tag{12.18}$$

The enthalpy of a species with respect to the reference enthalpy at standard pressure and temperature is defined as

$$h_k = \underbrace{\int_{T_0}^{T} c_{pk}\, \mathrm{d}T}_{\text{sensible}} + \underbrace{\Delta h_0^k}_{\text{chemical}} \tag{12.19}$$

where Δh_0^k is the enthalpy of formation (J/kg) of the species k at standard pressure and temperature. It is related to the enthalpy of formation Δh_f^k (J/kmol) introduced earlier by means of $\Delta h_0^k = \Delta h_f^k/(MW)_k$.

Properties such as the specific enthalpy h_{mix} of a mixture may be calculated from one of the following two expressions:

$$h_{mix} = \sum_{k}^{\text{all species}} Y_k h_k \text{ or} \tag{12.20}$$

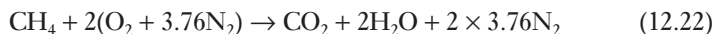
$$\bar{h}_{mix} = \sum_{k}^{\text{all species}} X_k \bar{h}_k \tag{12.21}$$

where h_k is the specific enthalpy (units J/kg) of species k and the overbar indicates molar specific enthalpy of species k, which has units J/kmol.

12.5 Stoichiometry

In combustion calculations we often perform a simple analysis which assumes that the fuel burns completely to form products (complete combustion). Most common combustion reactions involve oxidation of hydrocarbon fuels that have a high content of carbon (C) and hydrogen (H). When combustion is complete all C atoms in the fuel are consumed to form CO_2 and all H atoms in the fuel become H_2O. Any other combustible elements such as sulphur (S) etc. also combine with O_2. A chemical equation for complete combustion could be written by considering an atom balance. This would yield the exact amount of oxidiser required for complete combustion: the so-called stoichiometric oxidiser requirement. Usually the oxidiser is air, therefore the exact air requirement for complete combustion is called the **stoichiometric air requirement** for complete combustion. We can also calculate the air to fuel ratio necessary for complete combustion, which is called the **stoichiometric air/fuel ratio**. The composition of air is considered to be 79% of N_2 and 21% of O_2 by volume. For each mole of O_2 in air there are 79/21 (= 3.76) moles of N_2. By mass the ratio is 23.3% of O_2 and 76.7% of N_2. The molecular weight of air is taken as 29 kg/kmol; 1 kmol of O_2 is contained in 100/21 kmol of air (i.e. 4.76 kmol of air).

Complete combustion of CH_4 for example is given by the equation

$$CH_4 + 2(O_2 + 3.76N_2) \rightarrow CO_2 + 2H_2O + 2 \times 3.76N_2 \tag{12.22}$$

The stoichiometric air fuel/ratio by mass is

$$(A/F)_{st} = \frac{m_{air}}{m_{fuel}}$$

$$= \frac{(MW)_{air} \times \text{number of kmol of } O_2 \times \text{number of kmol of air per kmol of } O_2}{(MW)_{CH_4} \times \text{number of kmol of } CH_4}$$

$$= \frac{29 \times 2 \times 4.76}{16 \times 1} = 17.255 \text{ kg air/kg fuel} \tag{12.23}$$

12.6 Equivalence ratio

Combustion with less air than the stoichiometric air requirement is fuel rich, and combustion with air in excess of the stoichiometric air requirement is called lean combustion. The equivalence ratio is used in combustion calculations to define the strength of a mixture with respect to the stoichiometric

mixture strength. Given a fuel we may calculate the stoichiometric air/fuel ratio as explained above; say the value is $(A/F)_{st}$. In the actual mixture the air fuel ratio may be $(A/F)_{actual}$. The **equivalence ratio** indicates the strength of a mixture with respect to the stoichiometric air/fuel ratio. It is defined as follows:

$$\phi = \frac{(A/F)_{st}}{(A/F)_{actual}} = \frac{(F/A)_{actual}}{(F/A)_{st}} \tag{12.24}$$

If the equivalence ratio of an air/fuel mixture is equal to unity, the mixture is stoichiometric. A value less than 1, i.e. $\phi < 1$, indicates a lean mixture, and if $\phi > 1$, the mixture is rich.

12.7 Adiabatic flame temperature

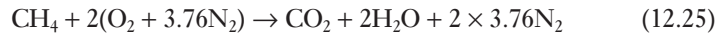
If a fuel/air mixture is considered to be burnt completely under constant pressure and if no external heat or work transfer takes place then all the energy liberated by the chemical reaction will be used to heat up the products. This process will achieve the maximum possible temperature, which is called the **adiabatic flame temperature**. It can be calculated using the first law of thermodynamics and enthalpies of reactants and products. When there are no heat losses the total enthalpy of products is equal to the enthalpy of reactants.

Example 12.1

Calculate the adiabatic flame temperature for combustion of a stoichiometric mixture of CH_4 and air at standard temperature and pressure.

Solution

The combustion equation is

$$CH_4 + 2(O_2 + 3.76N_2) \rightarrow CO_2 + 2H_2O + 2 \times 3.76N_2 \tag{12.25}$$

Application of the first law gives

$$H_{P2} - H_{R1} = (H_{P2} - H_{P0}) + (H_{P0} - H_{R0}) + (H_{R0} - H_{R1}) \tag{12.26}$$

For a constant pressure adiabatic process (neglecting kinetic energy changes)

$$H_{P2} - H_{R1} = 0 \tag{12.27}$$

It is given that the reactants start off at standard pressure and temperature so no heat is taken up in heating the reactants to bring them to standard temperature. Thus $H_{R0} - H_{R1} = 0$, and we can simplify equation (12.26) as

$$(H_{P2} - H_{P0}) = -(H_{P0} - H_{R0}) \tag{12.28}$$

$$\sum_{k}^{\text{all products}} m_k c_{pk}(T_2 - T_0) = -(H_{P0} - H_{R0}) \tag{12.29}$$

Here temperature T_2 is the final temperature of the products and $H_{P0} - H_{R0} = \Delta H_0$ is the heat of combustion of CH_4.

Before we calculate the value of T_2 we need to consider that the change in temperature during combustion will cause the values of specific heats c_v and c_p to change. In simple adiabatic flame temperature calculations mean specific heat values are used. To obtain mean c_{pk}-values for products (CO_2 and H_2O) we need a temperature range. In a calculation like this we start by assuming a final temperature to estimate a mean temperature for products and then

repeat the calculation using the estimated product temperature after the first calculation. We guess a final temperature T_2, say 2000 K. Then a mean temperature for products would be (2000 + 298)/2 = 1149 K. For gases at different temperatures the value of c_{pk} may be calculated using an equation of the form $c_p = a + bT + cT^2 + dT^3$ (see Cengel and Boles (2002) for values of the constants a, b, c and d):

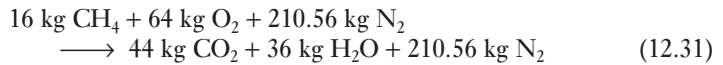
$$c_p \text{ value for } CO_2 \text{ at } 1149 \text{ K} = 1.274 \text{ kJ/kg K} \tag{12.30a}$$

$$c_p \text{ value for } H_2O \text{ at } 1149 \text{ K} = 2.384 \text{ kJ/kg K} \tag{12.30b}$$

$$c_p \text{ value for } N_2 \text{ at } 1149 \text{ K} = 1.1931 \text{ kJ/kg K} \tag{12.30c}$$

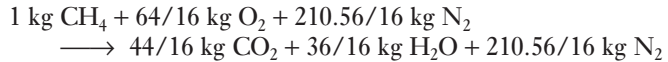
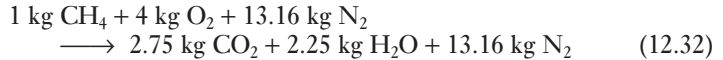
To evaluate T_2 from equation (12.29) we need the mass m_k of each of the product species. The combustion equation in mole form (12.25) can be converted into mass form by multiplying the molecular weight of each substance by the coefficient of the molar balance equation.

In terms of mass,

$$\begin{aligned} &16 \text{ kg } CH_4 + 64 \text{ kg } O_2 + 210.56 \text{ kg } N_2 \\ &\quad \longrightarrow 44 \text{ kg } CO_2 + 36 \text{ kg } H_2O + 210.56 \text{ kg } N_2 \end{aligned} \tag{12.31}$$

For CH_4 combustion the heat of combustion $H_{P0} - H_{R0} = -50\,050$ kJ/kg. We use the so-called lower heating value, which assumes that H_2O in the products is in the vapour phase. Since heat of combustion values are tabulated per kg of fuel we rewrite (12.31) as follows to obtain values for the mass of products species m_k:

$$\begin{aligned} &1 \text{ kg } CH_4 + 64/16 \text{ kg } O_2 + 210.56/16 \text{ kg } N_2 \\ &\quad \longrightarrow 44/16 \text{ kg } CO_2 + 36/16 \text{ kg } H_2O + 210.56/16 \text{ kg } N_2 \end{aligned}$$

or

$$\begin{aligned} &1 \text{ kg } CH_4 + 4 \text{ kg } O_2 + 13.16 \text{ kg } N_2 \\ &\quad \longrightarrow 2.75 \text{ kg } CO_2 + 2.25 \text{ kg } H_2O + 13.16 \text{ kg } N_2 \end{aligned} \tag{12.32}$$

We use equation (12.29) with the c_{pk}-values of (12.30a–c) and the values of mass of products species m_k from (12.32) and obtain

$$\begin{aligned} \sum m_k c_{pk}(T_2 - T_0) &= [(2.75 \times 1.274) + (2.25 \times 2.384) \\ &\quad + (13.16 \times 1.193)] \times 10^3 \times (T_2 - 298) \\ &= 50\,050 \times 10^3 \end{aligned}$$

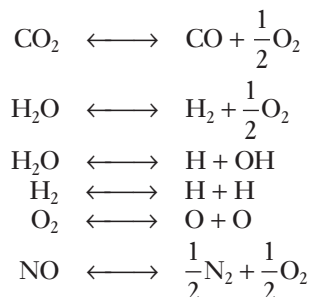
which gives $T_2 = 2335$ K.

As we have used an estimated T_2 at the start to obtain c_p, the calculation has to be repeated with the new T_2 to obtain a new mean value of temperature to determine a new set of mean c_p-values. Another iteration using this new mean temperature of 1316 K for properties yields a flame temperature of 2276 K. After a few more iterations we obtain a final value of the adiabatic flame temperature as 2288 K. This sort of calculation can be easily performed using an iterative spreadsheet with polynomial fits for c_p-values of the species. More precise calculations require integrals for mean c_p which could also be incorporated into a calculation using polynomial fits for c_p-values. Knowing the adiabatic flame temperature for a given air/fuel ratio is important in CFD, because it constitutes an upper limit for predicted temperatures in a combustion calculation. If a combustion calculation produces higher temperatures than the adiabatic flame temperature, the procedure has to be checked carefully and corrected. Sometimes such overpredictions can occur

due to underflow and overflow errors in the computations and the computational procedure should be structured to avoid such errors.

12.8 Equilibrium and dissociation

The above analysis for adiabatic flame temperature is based on complete combustion. In practice, however, at high temperatures some reactions occur in the reverse direction. This phenomenon is called dissociation. Dissociation reactions are endothermic: therefore, the actual temperature will be lower than the adiabatic flame temperature based on complete combustion. At typical combustion temperatures important dissociation equations are

$$\begin{aligned} CO_2 &\longleftrightarrow CO + \frac{1}{2}O_2 \\ H_2O &\longleftrightarrow H_2 + \frac{1}{2}O_2 \\ H_2O &\longleftrightarrow H + OH \\ H_2 &\longleftrightarrow H + H \\ O_2 &\longleftrightarrow O + O \\ NO &\longleftrightarrow \frac{1}{2}N_2 + \frac{1}{2}O_2 \end{aligned}$$

As a consequence of dissociation, species like CO, H_2, OH, O_2 will be present in the products. These are called minor species since they usually occur in much smaller concentrations than the main reaction products CO_2 and H_2O. Later we will see that combustion mechanisms for even simple fuels are much more complicated and result in many minor species in the products. The assumption of complete combustion is certainly a simplification. When concentrations of minor species in the products are known the adiabatic flame temperature of such a mixture can be calculated by taking into consideration all species of reactants and products.

Given the conditions for combustion and the air/fuel ratio we would like to know what the resulting composition of the mixture is after dissociation. When a mixture has reached chemical equilibrium, the composition of the gas mixture can be obtained using a parameter known as the equilibrium constant K_p. The equilibrium criterion is based on the second law of thermodynamics. See standard thermodynamics textbooks for the theory on the applications of the second law to combusting systems. In the analysis the thermodynamic property known as the Gibbs function is used. The Gibbs function is defined as

$$g = h - Ts \tag{12.33}$$

where h is specific enthalpy (units J/kg), T is temperature in K and s is specific entropy (units J/kg.K). The units of specific Gibbs function g are the same as the specific enthalpy, i.e. J/kg, and its values are available in thermodynamic tables (Rogers and Mayhew, 1994).

It can be shown that, as a consequence of the second law, a system seeks to maximise its entropy, so a reacting system is in equilibrium when the total Gibbs function (units J) attains a minimum value. Mathematically, the criterion is expressed in terms of

$$(\mathrm{d}G)_{T,p} = 0 \tag{12.34}$$

When this is applied to a combustion reaction in the form

$$aA + bB \longleftrightarrow cC + dD \tag{12.35}$$

where a, b, c and d are stoichiometric coefficients of the species participating in the reaction, and these species are perfect gases and the reaction is isothermal, it is possible to show that the **condition for equilibrium** is (see e.g. Kuo, 2005)

$$\Delta G_T^o = -R_u T \ln K_p \tag{12.36}$$

where ΔG_T^o is called the standard state Gibbs function change, K_p is the equilibrium constant and R_u is the universal gas constant.

Furthermore, it can be shown that

$$K_p = \frac{p_C^c p_D^d}{p_A^a p_B^b}$$

where p_A, p_B, p_C etc. are partial pressures of species A, B, C etc.

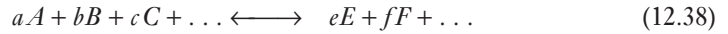
We note that each partial pressure is raised to a power corresponding to the stoichiometric coefficient. In equation (12.36) the **standard state Gibbs function change** ΔG_T^o may be calculated from

$$\Delta G_T^o = \Delta H^o - T\Delta S^o \tag{12.37}$$

These quantities may be obtained from thermodynamic property tables.

From equation (12.36) $K_p = \exp(-\Delta G_T^o / R_u T)$, so K_p-values may be obtained from thermodynamic property tables or calculated from the standard state Gibbs function change using equations (12.37) and (12.36).

If there are many reactants and products involved in the mixture in the form

$$aA + bB + cC + \ldots \longleftrightarrow \quad eE + fF + \ldots \tag{12.38}$$

the **equilibrium constant** K_p is given by

$$K_p = \frac{p_E^e p_F^f \ldots}{p_A^a p_B^b \ldots} = \frac{X_E^e X_F^f \ldots}{X_A^a X_B^b \ldots} \tag{12.39}$$

In the equation for K_p products appear in the numerator and reactants appear in the denominator. The equation also shows that partial pressures can be written in terms of mole fractions.

In practical equilibrium combustion calculations many independent reactions are involved, so we solve balance equations for overall mass or mole conservation along with expressions written for K_p in terms of mole fractions for each independent reaction to give a set of relationships between product and reactant mole fractions. Additional relationships are obtained from atom balances for each of the atomic species participating in the reaction and from the air/fuel ratio. This set of algebraic equations is solved to obtain mole fractions of all the molecular species. An illustrative example of this type of equilibrium chemistry calculation is given below.

Dedicated computer codes based on the principle of Gibbs function minimisation are used for equilibrium calculations in systems with complex reaction mechanisms involving many species (see e.g. Morley, 2005; Kuo, 2005). Further literature on equilibrium calculations can be found in Turns (2000) and Warnatz *et al.* (2001). The interested reader is also referred to the NASA equilibrium calculation program documented by Gordon and

McBride (1994) and available via the Internet at the CEA website (see URL under McBride, 2004).

Example 12.2

Methane and air at an equivalence ratio of $\phi = 1.25$ burn at 1600 K and 1.0 atm pressure. Determine the composition of the products, which include H_2 and CO.

Solution

The stoichiometric combustion equation is

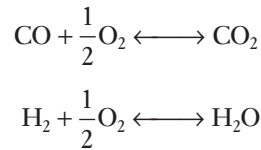
$$CH_4 + 2(O_2 + 3.76N_2) \rightarrow CO_2 + 2H_2O + 2 \times 3.76N_2 \quad (12.40)$$

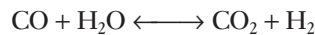
Equation (12.23) gives the stoichiometric air/fuel ratio $(A/F)_{st} = 17.255$. Equation (12.24) allows us to calculate the actual air/fuel ratio for the given equivalence ratio $\phi = 1.25$:

$$(A/F) = \frac{(A/F)_{st}}{\phi} = \frac{17.255}{1.25} = 13.804$$

The actual combustion process includes dissociation. The following dissociation reactions are responsible for the creation of the species H_2 and CO:

$$CO + \frac{1}{2}O_2 \longleftrightarrow CO_2$$

$$H_2 + \frac{1}{2}O_2 \longleftrightarrow H_2O$$

When the fuel mixture is richer than stoichiometric ($\phi > 1$) it can be argued that O_2 is very low as it will be consumed by H_2. The two chemical equations above can be combined into

$$CO + H_2O \longleftrightarrow CO_2 + H_2$$

This is called the water–gas equilibrium equation.

The *actual* combustion equation should therefore include these additional product species H_2 and CO:

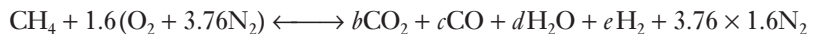
$$CH_4 + a(O_2 + 3.76N_2) \longleftrightarrow bCO_2 + cCO + dH_2O + eH_2 + 3.76aN_2$$

The number of moles of air a can now be calculated from the air/fuel ratio:

$$(A/F) = \frac{a \times 4.76 \times 29}{1 \times 16} = 13.804$$

$$a = 1.6$$

Thus,

$$CH_4 + 1.6(O_2 + 3.76N_2) \longleftrightarrow bCO_2 + cCO + dH_2O + eH_2 + 3.76 \times 1.6N_2$$

An atom balance for all three atomic species that participate in the combustion reaction yields three equations:

$$\text{Carbon (C) balance:} \quad 1 = b + c \quad (12.41)$$

$$\text{Oxygen (O) balance:} \quad 3.2 = 2b + c + d \quad (12.42)$$

$$\text{Hydrogen (H) balance:} \quad 4 = 2d + 2e \quad (12.43)$$

Since there are four unknown b, c, d and e we need a further equation. As we have seen, the equilibrium constant (12.39) connects the mole fractions of reactants and products, so the equilibrium constant K_p supplies the fourth equation.

K_p for the water–gas equilibrium equation is

$$K_p = \frac{p_{CO2} p_{H_2}}{p_{H_2O} p_{CO}} = \frac{(b/n)(e/n)}{(c/n)(d/n)} \tag{12.44}$$

$$K_p = \frac{be}{cd} \tag{12.45}$$

where n is the total number of moles in the mixture.

Values for K_p for the water–gas equilibrium equation are tabulated as a function of temperature in thermodynamic data tables (Rogers and Mayhew, 1994; or the JANAF tables in Chase *et al.*, 1985). For a combustion temperature of 1600 K, the tables show $\ln(K_p) = -1.091$, which gives $K_p = 0.3358$ and, hence,

$$0.3358 = \frac{be}{cd} \tag{12.46}$$

Now equations (12.41)–(12.43) and (12.46) are solved to obtain values of b, c, d and e. In this case by simple elimination a quadratic equation for b can be obtained

$$0.6642b^2 + 0.8746b - 0.7387 = 0$$

The solution gives

$$b = 0.5848$$
$$c = 0.4152$$
$$d = 1.6152$$
$$e = 0.3848$$

As mentioned earlier, dissociation reactions are endothermic and result in incomplete combustion. At equilibrium, reactants such as CO and H_2 are present in the products. This means that the fuel has not been utilised in full and the flame temperature will be lower than the adiabatic flame temperature estimated on the basis of complete combustion. However, it takes time for combustion products to achieve equilibrium. In practice, therefore, equilibrium calculations will give good predictions of product composition if the residence time is long. For example, in long furnaces which operate at high temperature the products might achieve equilibrium.

In many other combustion systems, where the flow or operating conditions do not provide adequate time for equilibrium to be reached, the product composition cannot be predicted from equilibrium alone. Other factors, such as finite rate reaction kinetics, turbulence, transient effects etc., have to be taken into account. The internal combustion engine is an important example of combustion taking place in a highly transient environment. Peak temperatures and pressures at which dissociation takes place occur only for a very short period. The rapid drop in pressures and temperatures immediately after the combustion phase causes the product composition to be 'frozen' before equilibrium is reached. Another important non-equilibrium phenomenon in engine combustion is pollutant formation, which is strongly influenced by chemical kinetics. These processes will be discussed in the next sections. Nevertheless, for many combustion systems equilibrium calculations give a fair representation of final products. Such details are very useful for the design of combustion equipment. Where applicable, CFD

calculations of combustion can be carried out with equilibrium chemistry models (Jones and Priddin, 1978; Nazha *et al*., 2001). For situations where equilibrium models are not applicable more elaborate combustion models are required. General-purpose CFD codes obviously need to include capabilities to deal with any situation, so composition calculations need to take into account effects such as departure from equilibrium, turbulence, finite rate chemistry and radiative heat transfer. The relevant modelling concepts are discussed in sections to follow.

12.9 Mechanisms of combustion and chemical kinetics

As mentioned earlier, combustion of a fuel does not occur in a single reaction, but may involve a number of different steps. In the above discussion it was mentioned that it takes time for products to reach equilibrium. Chemical kinetics ultimately determines how long it takes for a system to reach its final equilibrium state. Essentially, chemical kinetics is the study of reaction mechanisms and reaction rates.

Many combustion processes are physically controlled, i.e. the rate of combustion depends on flow, turbulence and diffusion processes. Some examples are: (i) wick flames such as candle flames and oil lamps, (ii) combustion of pools of liquids, pool fires, (iii) droplet combustion, burning of liquid fuels in furnaces, (iv) diesel engine combustion, (v) gas turbine combustion at low altitude, (vi) rocket motors, (vii) laminar and turbulent jet diffusion flames, (viii) combustion in boilers and furnaces, (ix) turbulent premixed combustion in petrol engines, aeroengines etc. In situations such as these diffusion and turbulence dominate mixing and subsequent combustion.

However, chemical kinetics can play an important role under certain conditions, e.g. when the pressure is low or the supply of oxygen is restricted. Combustion is never independent of physical processes, so 'kinetically influenced' is the correct way to describe these processes. Some examples of kinetically influenced situations are: (i) propagation of laminar flames through premixed fuel and oxidant, (ii) aerated laminar flames such as gas cooker and domestic boiler flames, (iii) ignition processes such as spark ignition in petrol engines, auto-ignition in diesel engines, ignition in domestic appliances, (iv) extinction processes such as extinction of gas turbines at high altitude, petrol engines running too weak or too rich, extinction of stabilised flames, (v) complex situations where kinetics competes with mixing, e.g. burning of very small droplets or particles, insufficient availability of oxygen in pulverised coal flames, situations with large radiative heat losses from coal particles and highly sheared flow situations.

12.10 Overall reactions and intermediate reactions

In the combustion calculations which we discussed earlier we wrote the combustion equation as a single overall reaction or global reaction:

$$\text{Fuel} + \text{Oxidant} \rightarrow \text{Products} \tag{12.47}$$

For example,

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O \tag{12.48}$$

In practice this reaction does not occur in this single-step fashion, since it would require the simultaneous meeting of three different reactant

molecules. Chemical reactions more commonly occur due to collisions of pairs of molecules, where chemical bonds are broken during impact and new bonds are formed. In the process many intermediate bonds and intermediate species are formed. Such reactions are called elementary reactions and involve stable intermediate species and radicals. For example, some common radicals involved in reactions are H, O, OH, CH, HO_2, C_2. These radicals are unstable and highly reactive because they contain unpaired electrons. The simplest and most well-documented reaction is H_2 combustion:

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O \tag{12.49}$$

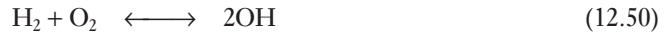
Even this simple reaction involves many intermediate steps such as

$$H_2 + O_2 \longleftrightarrow 2OH \tag{12.50}$$

$$H_2 + OH \longleftrightarrow H_2O + H \tag{12.51}$$

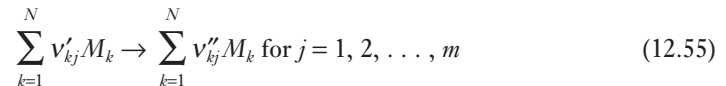
$$H + O_2 \longleftrightarrow OH + O \tag{12.52}$$

$$H_2O + O \longleftrightarrow OH + OH \tag{12.53}$$

$$H + OH + M \longleftrightarrow H_2O + M \tag{12.54}$$

The symbol 'M' represents a third body that can be any of the species present in the system. The subject of chemical kinetics deals with details of how these reactions occur. There are many mechanisms and paths of reactions, and chemical kinetics has been a research area which has attracted a great deal of attention. We do not intend to go on into the details here, and the reader is referred to Turns (2000), Kuo (2005), Warnatz *et al.* (2001), Bartok and Sarofim (1991), Gardiner (1984) and further references therein for more details. Below we outline some of the basic concepts useful for CFD combustion procedures.

12.11 Reaction rate

In combustion modelling it is required to determine rates of reactant consumption and product formation. These are used as source terms in transport equations for each of the species. It has already been mentioned that many intermediate equations are involved in combustion, and one particular species may be formed and consumed in a number of different reactions where some of these reactions may be reversible. For illustration purposes let us consider a scheme of (forward-only) reactions represented by the stoichiometric equation

$$\sum_{k=1}^{N} \nu'_{kj} M_k \rightarrow \sum_{k=1}^{N} \nu''_{kj} M_k \text{ for } j = 1, 2, \ldots, m \tag{12.55}$$

ν'_{kj} are stoichiometric coefficients of reactant species M_k in the reaction j. ν''_{kj} are stoichiometric coefficients of product species M_k in the reaction j. N is the total number of species involved and m is the total number of reactions in the scheme.

Mass conservation enforces that

$$\sum_{k=1}^{N} \nu'_{kj} (MW)_k = \sum_{i=1}^{N} \nu''_{kj} (MW)_k \text{ for } j = 1, 2, \ldots, m \tag{12.56}$$

For example, consider the simple reaction

$$2H_2 + O_2 \rightarrow 2H_2O \quad (12.57)$$

For this reaction the total number of species involved is $N = 3$; the species involved are $M_1 = H_2$, $M_2 = O_2$ and $M_3 = H_2O$.

Considering the left hand side of equation (12.57) we have

Stochiometric coefficient ν_1' for $H_2 = 2$
Stochiometric coefficient ν_2' for $O_2 = 1$
Stochiometric coefficient ν_3' for $H_2O = 0$, no H_2O on the left hand side of the equation

Considering the right hand side of equation (12.57) we have

Stochiometric coefficient ν_1'' for $H_2 = 0$, no H_2 on the right hand side of the equation
Stochiometric coefficient ν_2'' for $O_2 = 0$, no O_2 on the right hand side of the equation
Stochiometric coefficient ν_3'' for $H_2O = 2$

The mass conservation (12.56) can be verified for this equation as

Left hand side $(2 \times 2 + 1 \times 32) = (2 \times 18)$ Right hand side

The law of mass action (see Kuo, 2005; Turns, 2000), which has been confirmed by experimental observations, states that the rate of consumption of a chemical species in a reaction is proportional to the (mathematical) product of the concentration of the reacting species, each concentration being raised to the powers corresponding to the stoichiometric coefficients of the reactants.

The **reaction rate** RR (units kmol/m^3.s) for the reaction (12.55) is given by (see Kuo, 2005)

$$\mathrm{RR} = \frac{\mathrm{d}(C_{product})}{\mathrm{d}t} = k_f \prod_{k=1}^{N} (C_{M_k})^{v'_{kj}} \quad (12.58)$$

Since species M_k appear on both sides of equation (12.55), the **net progress rate of reaction** j is given by the production of the species M_k minus the destruction of the species M_k:

$$\dot{q}_{kj} = \frac{\mathrm{d}C_{M_k}}{\mathrm{d}t} = (\nu_k'' - \nu_k') k_f \prod_{k=1}^{N} (C_{M_k})^{v'_{kj}} \quad (12.59)$$

Here C denotes the molar concentration of the species in units kmol/m^3 (or gmol/cm^3 in CGS units) and the proportionality constant k_f is called the specific reaction rate constant. It should be noted here that the subscript k used to identify species is different from this k. The reaction rate constant k_f is independent of the concentration and is usually expressed as

$$k_f = AT^{\alpha} \exp\left(-\frac{E_a}{R_u T}\right) \quad (12.60)$$

This is known as the **Arrhenius law**. In this expression A is called the pre-exponential constant. Parameter α is a temperature exponent and E_a is the activation energy. These constants are specific to a given reaction. R_u is the universal molar gas constant (= 8.314 kJ/kmol.K) and T is absolute

temperature. A note about units is required here. Consider the simple forward-only reaction

$$aA + bB \rightarrow cC + dD \tag{12.61}$$

The progress rates of reactant consumption and product formation for this reaction can be evaluated using equation (12.59) as

$$\dot{q}_A = \frac{\mathrm{d}C_A}{\mathrm{d}t} = -ak_f(C_A)^a(C_B)^b \tag{12.62}$$

$$\dot{q}_B = \frac{\mathrm{d}C_B}{\mathrm{d}t} = -bk_f(C_A)^a(C_B)^b \tag{12.63}$$

$$\dot{q}_C = \frac{\mathrm{d}C_C}{\mathrm{d}t} = ck_f(C_A)^a(C_B)^b \tag{12.64}$$

$$\dot{q}_D = \frac{\mathrm{d}C_D}{\mathrm{d}t} = dk_f(C_A)^a(C_B)^b \tag{12.65}$$

Note that equation (12.61) is a forward-only reaction. When a backward reaction is also present similar expressions can be written using a backward reaction rate constant k_b. The units of C_A and C_B are kmol/m^3. The units of the progression rate are kmol/m^3.s. Because of the involvement of exponents in equations (12.59) it can be seen that the units of k_f, which is given by equation (12.60), depend on the stoichiometry of the reaction. For the reaction (12.61), the sum of the coefficients a and b is called the reaction order $n = a + b$. E_a has units kJ/kmol. RT has the same units. The exponential term $-E_a/RT$ does not contribute to units. The factor T^α has units depending on the value of α (see Turns, 2000). Therefore for this particular reaction the constant A has units $(\mathrm{m}^3/\mathrm{kmol})^{n-1}/(\mathrm{K}^{-\alpha}\ \mathrm{s})$. The units of k_f are $(\mathrm{m}^3/\mathrm{kmol})^{n-1}/\mathrm{s}$. In most cases α is zero, in which case the units required for k_f are the same as those for A (see e.g. Henson and Salimian, 1984).

For a simple reversible reaction of the form

$$aA + bB \longleftrightarrow cC + dD \tag{12.66}$$

the progress rate of consumption of the reactant species A in equation (12.66) is

$$\dot{q}_A = \frac{\mathrm{d}C_A}{\mathrm{d}t} = -a[k_f(C_A)^a(C_B)^b - k_b(C_C)^c(C_D)^d] \tag{12.67}$$

Similarly for a large scheme of reactions of the form

$$\sum_{k=1}^{N} \nu'_{kj} M_k \longleftrightarrow \sum_{k=1}^{N} \nu''_{kj} M_k \quad \text{for } j = 1, 2, \ldots, m \tag{12.68}$$

the most general form of the reaction progress rate for species k by reaction j is given by

$$\boxed{\dot{q}_{kj} = \left.\frac{\mathrm{d}C_{M_k}}{\mathrm{d}t}\right|_j = (\nu''_k - \nu'_k) \cdot \left[k_f \prod_{k=1}^{N} (C_{M_k})^{\nu'_{kj}} - k_b \prod_{k=1}^{N} (C_{M_k})^{\nu''_{kj}}\right]} \tag{12.69}$$

where k_f is the forward reaction rate constant and k_b is the backward reaction rate constant for the reaction j. This kind of compact notation is very useful in handling large systems of kinetic schemes containing many equations (more than 100 or so) and is used in common kinetic solution algorithms such as CHEMKIN (Kee *et al*., 1996).

In complex chemical schemes involving m equations the total rate of production rate $\dot{q}_k$ of a certain species k is the sum of individual rates of each equation producing the species k, i.e.

$$\boxed{\dot{q}_k = \sum_{j=1}^{m} \dot{q}_{kj}} \tag{12.70}$$

where $\dot{q}_k$ is the **total reaction rate for the species** k and $\dot{q}_{kj}$ is the reaction rate for species k for the specific reaction j.

These reaction rates are used in the transport equations for each of the species in the solution of combustion problems. When reaction rate expressions are written for each species participating in the mechanism this yields a system of first-order differential equations describing the progress of a chemical system with time. Given the initial conditions (initial concentration of the reacting species and temperature), additional equations for conservation of mass, momentum and energy can be written, and the system of differential equations can be numerically integrated to obtain a solution. In the solution of such systems, when one or more variables change rapidly in comparison with other variables, the resulting system is said to be stiff. Chemical kinetic solution packages use special routines to deal with such stiff systems (see Kee *et al*., 1996; Radhakrishnan and Pratt, 1988).

The reaction rates are used as source terms (see later) in transport equations for species mass fractions, which have the units kg/m^3.s. To convert to required units we multiply $\dot{q}_{kj}$, which is in kmol/m^3.s, by the molecular weight (MW). After converting to mass rate units the usual reaction rate **source term used in species transport equations** may be written as

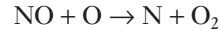
$$\boxed{\dot{\omega}_k = (MW)_k \dot{q}_k} \tag{12.71}$$

where $\dot{q}_k$ is obtained from (12.70) and (12.69) using details of kinetic data (values of constants A, α, E_a etc.) from Baulch *et al.* (1994), Turns (2000), Smith *et al.* (2003), Kuo (2005), Gardiner (1984) and many other publications cited later. In some of the early combustion literature tabulated kinetic data have been tabulated in CGS units. Care must be given in using these datasets, and appropriate SI unit conversion is required when incorporating reaction rates.

A further point to note is the link between forward and backward reaction rate constants. When the forward reaction rate constant (k_f) is known, the backward rate constant (k_b) for the jth reaction can be calculated through the equilibrium constant introduced in section 12.8:

$$\boxed{k_{fj} = k_{bj}(C_{M_j})^{\sum_{k=1}^{N}(\nu''_{kj}-\nu'_{kj})} = \frac{\prod_{\text{products}} X_i^{\nu''_i}}{\prod_{\text{reactants}} X_i^{\nu'_i}} = K_P(R_u T/p)^{\sum \nu'_i - \sum \nu''_i}} \tag{12.72}$$

For example, one of the reactions known as the Zel'dovich mechanism for thermal NO formation (discussed later) is

$$NO + O \rightarrow N + O_2$$

The rate coefficient (from Turns, 2000) for the forward reaction is

$$k_f = 3.80 \times 10^9 T^{1.0} \exp(-20820/T) \text{ cm}^3/\text{gmol.s}$$

At 2300 K, the value is $k_f = 1.029 \times 10^9$ cm^3/gmol.s. For this reaction $(\sum \nu_i' - \sum \nu_i'') = 0$. The K_P-value for the above reaction at 2300 K is $K_P = 1.94 \times 10^{-4}$. Using equation (12.72) the backward reaction rate coefficient is obtained as

$$k_b = k_f/K_P = \frac{1.024 \times 10^9}{1.94 \times 10^{-4}} = 5.278 \times 10^{12} \text{ cm}^3/(\text{gmol s})$$

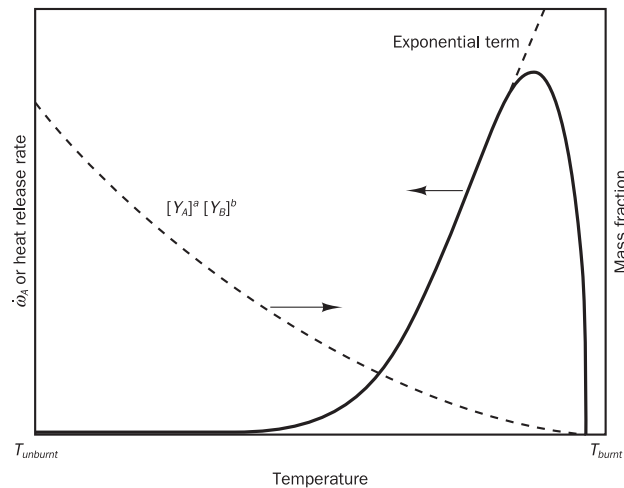
Further details of chemical kinetics, the fundamental development of reaction rate expressions, reaction kinetic data and how they are used in packages like CHEMKIN are available in Turns (2000) and Kuo (2005).

To illustrate the behaviour of the reaction rate expression for a simple reaction of the form (12.61) the rate of consumption of species A can be written as

$$\boxed{\dot{\omega}_A = A'[Y_A]^a[Y_B]^b \exp(-E/R_u T)} \tag{12.73}$$

where $A' = (MW)_A\, a\, AT^{\alpha}\left(\frac{\rho}{(MW)_A}\right)^a \left(\frac{\rho}{(MW)_B}\right)^b$

It should be noted that constant A' contains density and molecular weight terms, which arise from the conversion from molar concentrations (12.69) to mass fractions (as used in CFD computations). According to this expression, it can be seen that the rate increases exponentially with temperature. As the reaction progresses, the mass fractions of reactants decrease as they get consumed. Therefore the reaction rate decreases as species A and B are consumed. The typical behaviour of the rate expression is shown in Figure 12.1.

Figure 12.1 Schematic representation of the reaction rate expression for a single-step reaction

The reaction rate increases almost exponentially with temperature (T) at first because of the dominant role of the term $\exp(-E/RT)$. As T increases towards the final temperature T_{burnt}, the reaction rate drops in spite of the exponential term because $[Y_A]^a[Y_B]^b$ decreases rapidly as reactants A and B species are consumed in the reaction. This demonstrates that the availability of reactants (concentrations) and temperature both play major roles in determining the reaction rate.

12.12 Detailed mechanisms

There are many types of reaction: reactions of various orders (first-order, second-order etc.), consecutive reactions, competitive reactions, opposing reactions, chain reactions, chain branching reactions etc. For oxidation of various fuels detailed reaction mechanisms and appropriate rate constants for those reactions are available in the literature. For example, see Gardiner (1984), Drake and Blint (1988), Dryer (1991), Smooke (1991), Peters (1993), Turns (2000), Seshadri and Williams (1994), Warnatz *et al.* (2001), the GRI 3.0 mechanism (Smith *et al.*, 2003 and references therein), and also the San Diego mechanism (2003, http://maemail.ucsd.edu/combustion/cermech/).

Detailed mechanisms for methanol and the $CO/H_2/O_2$ system are available in Dryer (1991). This reference also gives one-step, two-step and four-step mechanisms for a variety of hydrocarbon fuels. Peters (1993) is a valuable source book which has contributions from many other authors and contains details of multi-step and reduced mechanisms for a range of fuels. A detailed mechanism for methane combustion, NO formation, single-step and multi-step reaction mechanisms for common hydrocarbons are also available in Turns (2000). A mechanism containing 46 steps for methane combustion has been reported in Smooke *et al.* (1986). A very well-known and widely used mechanism for methane combustion is GRI 2.11 (Bowman *et al.*, 1996). The most recent version of this mechanism is GRI 3.0 (Smith *et al.*, 2003), which includes a detailed mechanism for NO formation. However, it has been observed in some test cases that the previous version GRI 2.11 appears to give better NO predictions (at the time of writing this text) than the latest version GRI 3.0 (see Kim and Huh, 2002). Further details of detailed mechanisms can be found in Seshadri and Williams (1994), Warnatz *et al.* (2001), and also at the Lawrence Livermore Laboratory mechanisms link (http://www-cms.llnl.gov/combustion/combustion2.html).

It should be noted that rate constants of various mechanisms have been derived on the basis of computational estimations and comparisons with measurable combustion properties such as flame speeds, temperatures and species mass fractions obtained in controlled experiments. For large systems of chemical equations dedicated computer programs are required to solve chemical kinetic problems. CHEMKIN (Kee *et al.*, 1996), for example, is a widely used software package for such problems, and many commercial CFD codes allow the incorporation of its information relating to chemical reactions.

12.13 Reduced mechanisms

The computational cost of chemical kinetics evaluations and the associated species transport equations is substantial, and grows rapidly as the reaction mechanisms become more elaborate and detailed. Consequently, efforts have

been made to develop more practical reaction schemes involving fewer reactions to represent combustion of basic fuels. These simplified schemes are called reduced mechanisms and consist of a few key equations to predict major and important minor species (see e.g. Dryer, 1991; Seshadri and Williams, 1994). For example, the detailed mechanism for H_2 oxidation given in Conaire *et al*. (2004) consists of 19 reactions. Massias *et al*. (1999) give a five-step reduced mechanism for H_2 combustion including NO formation (Table 12.1).

Table 12.1

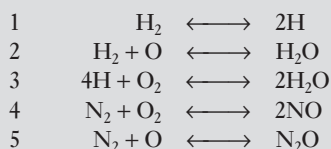
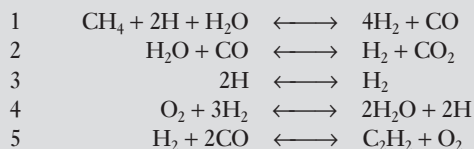
1	H_2	$\longleftrightarrow$	$2H$
2	$H_2 + O$	$\longleftrightarrow$	H_2O
3	$4H + O_2$	$\longleftrightarrow$	$2H_2O$
4	$N_2 + O_2$	$\longleftrightarrow$	$2NO$
5	$N_2 + O$	$\longleftrightarrow$	N_2O

The detailed chemical–kinetic mechanism for methane combustion shown in Seshadri and Williams (1994) consists of 39 elementary reactions involving 17 chemical species including N_2. A five-step reduced mechanism by Hewson and Bollig (1996) for CH_4 combustion is shown in Table 12.2. The same reference gives a 52-step, 13-species reaction detailed mechanism for nitrogen chemistry (species NO, NH_3, HCN etc.) and a reduced six-step reduced mechanism for nitrogen chemistry.

Table 12.2

1	$CH_4 + 2H + H_2O$	$\longleftrightarrow$	$4H_2 + CO$
2	$H_2O + CO$	$\longleftrightarrow$	$H_2 + CO_2$
3	$2H$	$\longleftrightarrow$	H_2
4	$O_2 + 3H_2$	$\longleftrightarrow$	$2H_2O + 2H$
5	$H_2 + 2CO$	$\longleftrightarrow$	$C_2H_2 + O_2$

Rate constants and further data for the reduced mechanisms shown in Tables 12.1 and 12.2 can be obtained from the above cited references. Extensive studies have been made of most practically important combustion reactions, resulting in a vast body of publications with proposals for detailed and reduced mechanisms. Some detailed mechanisms involve many equations – the above-mentioned GRI 3.0 mechanism for natural gas combustion including NO_x chemistry, for example, consists of 325 reactions and 53 species. The accuracy of reduced mechanisms has been compared with detailed mechanisms for many practically important combustion reactions (see e.g. Barlow *et al*., 2001; Massias *et al*., 1999). The choice of a reduced mechanism is always a trade-off between computational cost and accuracy required. The user should carefully select a suitable mechanism depending on the particular application and available resources. When results are interpreted it should also be noted that some reduced mechanisms have been tuned for specific temperature ranges and specific types of combustion processes.

12.14 Governing equations for combusting flows

So far we have discussed thermodynamics, chemical equilibrium and chemical kinetics without any reference to flow conditions. In many combustion situations fluid flow is an integral part of the combustion process. In non-premixed combustion situations such as furnaces, fuel and air streams are mixed by fluid flow and turbulence, and the resulting combustion temperatures, species concentrations and distribution of species are very much controlled by fluid flow. Similarly, in premixed combustion, e.g. a spark ignition IC engine, the geometry, fluid flow and turbulence created by the induction and compression processes prior to ignition play an important role in the combustion characteristics of the engine. We now present the transport equations governing gaseous fuel combustion. We quote all equations in compact suffix notation, introduced first in Chapter 3.

The governing equations of continuity and momentum developed in Chapter 2 can be used in unchanged form.

Continuity

The continuity equation is

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0 \tag{12.74}$$

It should be noted that density in combusting flows is a variable, and depends on pressure, temperature and species concentration.

Momentum equations

As in other flows the velocity field is governed by momentum equations

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_i}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_i} + F_i \tag{12.75}$$

where τ_{ij} is the viscous stress tensor and F_i is the body force (which includes gravity):

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\right) \tag{12.76}$$

The first transport equation that is specific to reacting flows is the equation for the conservation of mass of a species k. It can be easily developed from the general transport equation (2.39) by setting $\phi = Y_k$. In suffix notation the statement is as follows.

Transport equations for species (k)

$$\frac{\partial}{\partial t}(\rho Y_k) + \frac{\partial}{\partial x_i}(\rho u_i Y_k) = \frac{\partial}{\partial x_i}\left(\rho D_k \frac{\partial Y_k}{\partial x_i}\right) + \dot{\omega}_k \tag{12.77}$$

Rate of change of mass of species k	+	Net rate of decrease of mass of species k due to convection	=	Net rate of increase of mass of species k due to diffusion	+	Net rate of increase of mass of species k due to sources

In equation (12.77) D_k is the species diffusion coefficient (units m^2/s). The volumetric rate of generation (or destruction) of a species due to chemical reactions appears as the source (or sink) term $\dot{\omega}_k$ in each of their transport equations. For kinetically controlled combustion this term takes the form of (12.71) or (12.73). A wide variety of different models are available for physically controlled combustion. The most important ones will be discussed in the remaining sections of this chapter.

It is common practice to assume a single diffusion coefficient for all species. This simplifies equation (12.77) to

$$\frac{\partial}{\partial t}(\rho Y_k) + \frac{\partial}{\partial x_i}(\rho u_i Y_k) = \frac{\partial}{\partial x_i}\left(\rho D \frac{\partial Y_k}{\partial x_i}\right) + \dot{\omega}_k \tag{12.78}$$

Whilst the single D assumption is not always accurate (and may in fact be quite inaccurate) it is very attractive, since it enables far-reaching simplification of combustion calculations.

Energy equation

In combusting flows the temperature depends on the thermodynamic state and the composition of the mixture. Some combustion models do not require a transport equation for enthalpy: for example, in the laminar flamelet model the temperature is obtained from the laminar flamelet library curves. Other combustion submodels, however, require the solution of the transport equation for enthalpy. Chemical energy is released as heat during combustion, and the resulting enthalpy is obtained by solving its transport equation:

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_i}(\rho u_i h) = \frac{\partial}{\partial x_i}\left[\frac{\mu}{\sigma_h}\frac{\partial h}{\partial x_i} + \mu\left(\frac{1}{Sc_k} - \frac{1}{\sigma_h}\right)\sum_{k=1}^{N} h_k \frac{\partial Y_k}{\partial x_i}\right] + \frac{\partial p}{\partial t} + S_{rad} \tag{12.79}$$

Rate of change of enthalpy	+	Net rate of decrease of enthalpy due to convection	=	Net rate of increase of enthalpy due to diffusion along gradients of enthalpy	+	Net rate of increase of enthalpy due to mass diffusion along gradients of species concentration	+	Net rate of increase of enthalpy due to pressure work	+	Net rate of increase of enthalpy due to radiative heat transfer

In equation (12.79) the source term of the transport equation for enthalpy S_{rad} is the radiation loss or gain. Viscous energy dissipation is normally assumed to be negligible in low Mach number combusting flows. Here h is the mixture enthalpy per unit mass and h_k is the specific enthalpy of species k, and the summation is carried out over all N species considered in the chosen reaction mechanism. The mixture Prandtl number is σ_h, and Sc_k is the species Schmidt number, $Sc_k \equiv \mu/\rho D_k$.

The Prandtl number (σ_h) is defined as

$$\sigma_h = \frac{c_p \mu}{k} = \frac{\text{Rate of momentum transport}}{\text{Rate of energy transport}}$$

The Lewis number is defined as

$$Le_k \equiv \frac{k}{\rho c_p D_k} = \frac{\text{Rate of energy transport}}{\text{Rate of mass transport}}$$

The Schmidt number is defined as

$$Sc_k = \frac{\mu}{\rho D_k} = \frac{\text{Rate of momentum transport}}{\text{Rate of mass transport}}$$

$$= Le\sigma_h$$

If a single diffusion coefficient is used, i.e. $D_k = D$ for $k = 1, 2, \ldots, N$, as in equation (12.78), equation (12.79) can be written as

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_i}(\rho u_i h) = \frac{\partial}{\partial x_i}\left[\frac{\mu}{\sigma_h}\frac{\partial h}{\partial x_i} + \frac{\mu}{\sigma_h}\left(\frac{\sigma_h}{Sc} - 1\right)\sum_{k=1}^{N} h_k \frac{\partial Y_k}{\partial x_i}\right] + \frac{\partial p}{\partial t} + S_{rad} \quad (12.80)$$

This can be rewritten as

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_i}(\rho u_i h) = \frac{\partial}{\partial x_i}\left[\frac{\mu}{\sigma_h}\frac{\partial h}{\partial x_i} + \frac{\mu}{\sigma_h}\left(\frac{1}{Le} - 1\right)\sum_{k=1}^{N} h_k \frac{\partial Y_k}{\partial x_i}\right] + \frac{\partial p}{\partial t} + S_{rad} \quad (12.81)$$

Therefore for the case where the Lewis number is unity the enthalpy equation simplifies to

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_i}(\rho u_i h) = \frac{\partial}{\partial x_i}\left[\frac{\mu}{\sigma_h}\frac{\partial h}{\partial x_i}\right] + \frac{\partial p}{\partial t} + S_{rad} \quad (12.82)$$

For low-speed flows $\partial p/\partial t$ can be neglected, so under the assumptions of low-speed flow, single diffusion coefficient and unity Lewis number, the enthalpy equation has exactly the same form as the general transport equation (2.39). Furthermore, if the radiation source term S_{rad} is also small, the enthalpy is a conserved or passive scalar.

It should be noted that the thermal conductivity of individual species is a function of temperature and can be evaluated using a polynomial expression in the form $k = a + bT + cT^2 + dT^3$, where coefficients are available in Reid *et al.* (1987), for example. The binary diffusion coefficient D_{ij} between two species depends on the concentration of components, temperature and pressure. Key equations which can be used to calculate D_{ij} and then a single diffusion coefficient D_k (diffusion coefficient of a single species to the rest of mixture) are also available in the above reference. D_k also depends on individual species concentrations, temperature and pressure. The reader is also referred to Paul and Warnatz (1998), who describe transport models for species properties. In combustion modelling the required diffusion coefficients and other transport properties are calculated (in detailed chemistry

models) using packages such as CHEMKIN (Kee *et al.*, 1996). Just to quote some rough values, thermal conductivities of H_2, H_2O and CO_2 at 1000 K are 0.4 W/mK, 0.097 W/mK and 0.068 W/mK respectively. Approximate values of binary diffusion coefficient for mixtures with N_2 at 1000 K reported in Paul and Warnatz (1998) are: for H–N_2, 10 cm^2/s; for OH–N_2, 2 cm^2/s; and for O_2–N_2, 1.5 cm^2/s. Lewis numbers for species in an H_2/air mixture in a one-dimensional laminar flame calculation evaluated at the maximum species mass fraction location are shown in Table 12.3. Apart from species H and H_2, other species have Lewis numbers close to 1.0. These values are presented for information only, and the actual values should be evaluated case by case depending on local conditions. The reader is also referred to Clarke (2002) for further details on the calculation of Lewis numbers for various fuel/air mixtures.

Table 12.3 Lewis numbers for species evaluated in H_2/air combustion

N_2	O_2	H	OH	O	H_2	H_2O	H_2O_2
0.920	1.052	0.200	0.726	0.700	0.175	0.995	1.005

Other relationships

The total of mass fractions of fuel, oxidant and inert species is equal to 1, so

$$\sum_{k}^{\text{all species}} Y_k = 1 \tag{12.83}$$

where k represents species.

The temperature can be calculated from the enthalpy by means of

$$T = \frac{h - \sum_{k}^{\text{fuel species}} Y_k \Delta h_{fk}}{\bar{c}_p} \tag{12.84}$$

where Δh_{fu} is the enthalpy of combustion.

The average value of the specific heat $\bar{c}_p$ is defined as follows:

$$\bar{c}_p = \frac{1}{(T - T_{ref})} \int_{T_0}^{T} c_p \, \mathrm{d}T \tag{12.85}$$

Here

$$c_p \equiv \sum_{k}^{\text{all species}} Y_k c_{p,k}$$

and $c_{p,k}$ is the specific heat of species k. Polynomial fits for temperature-dependent $c_{p,k}$ for various species are available in various texts: see for example Cengel and Boles (2002).

The local density of the mixture is dependent on pressure, reactant and product concentrations and on the mixture temperature. Its value can be calculated from the following equation of state:

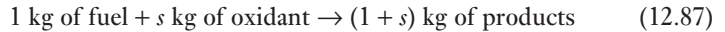
$$\rho = \frac{p}{R_u T \sum_k^{\text{all species}} \frac{Y_k}{(MW)_k}} \tag{12.86}$$

where $(MW)_k$ is the molecular weight of species k and R_u is the universal gas constant (8.314 kJ/kmol.K).

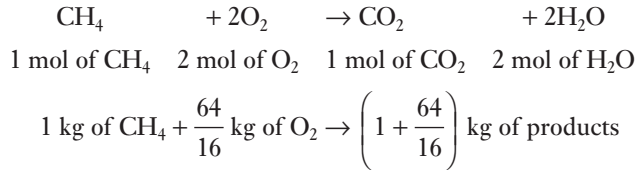
The flow field is in turn affected by changes in temperature and density, so in addition to the species and enthalpy equations we must solve all the flow equations. The resultant set of transport equations can be very large. Models that consider many intermediate reactions require vast computing resources, so simple models that incorporate only a few reactions are often preferred in CFD-based combustion procedures. The simplest known procedure is the simple chemical reacting system (SCRS; see Spalding, 1979), which is described below in some detail. Other approaches of modelling turbulent combustion, such as the eddy break-up model and the laminar flamelet model, are discussed later.

12.15 The simple chemical reacting system (SCRS)

If we are concerned with the global nature of the combustion process and with final major species concentrations only, the detailed kinetics is unimportant and a global one-step, infinitely fast, chemical reaction can be assumed where oxidant combines with fuel in stoichiometric proportions to form products:

$$1 \text{ kg of fuel} + s \text{ kg of oxidant} \rightarrow (1+s) \text{ kg of products} \tag{12.87}$$

For methane combustion the equation becomes

$$\begin{array}{cccc} CH_4 & +\,2O_2 & \rightarrow CO_2 & +\,2H_2O \\ 1 \text{ mol of } CH_4 & 2 \text{ mol of } O_2 & 1 \text{ mol of } CO_2 & 2 \text{ mol of } H_2O \end{array}$$

$$1 \text{ kg of } CH_4 + \frac{64}{16} \text{ kg of } O_2 \rightarrow \left(1 + \frac{64}{16}\right) \text{ kg of products}$$

The stoichiometric oxygen/fuel ratio by mass s is equal to 64/16 = 4 for methane combustion. However, equation (12.87) also shows that the rate of consumption of fuel during stoichiometric combustion is $1/s$ times the rate of consumption of oxygen, i.e.

$$\dot{\omega}_{fu} = \frac{1}{s}\dot{\omega}_{ox}$$

In the SCRS infinitely fast chemical reactions are assumed and the intermediate reactions are ignored. The transport equations for fuel and oxygen mass fraction are written as

$$\frac{\partial(\rho Y_{fu})}{\partial t} + \operatorname{div}(\rho Y_{fu}\mathbf{u}) = \operatorname{div}(\Gamma_{fu} \operatorname{grad} Y_{fu}) + \dot{\omega}_{fu} \tag{12.88}$$

$$\frac{\partial(\rho Y_{ox})}{\partial t} + \operatorname{div}(\rho Y_{ox}\mathbf{u}) = \operatorname{div}(\Gamma_{ox} \operatorname{grad} Y_{ox}) + \dot{\omega}_{ox} \tag{12.89}$$

where $\Gamma_{fu} = \rho D_{fu}$ and $\Gamma_{ox} = \rho D_{ox}$.

The oxidant stream will have nitrogen which remains unaffected during simple combustion; the mass fraction of inert species (Y_{in}) remains the same before and after combustion. Local values of Y_{in} are determined by mixing only, since the inert species does not take part in combustion (unless formation of NO is considered). Since the mass fraction of products $Y_{pr} = 1 - (Y_{fu} + Y_{ox} + Y_{in})$ it is unnecessary to solve a separate equation for Y_{pr}.

It is possible to reduce the number of transport equations even further by introducing a variable defined as follows:

$$\phi = sY_{fu} - Y_{ox} \tag{12.90}$$

Application of the single diffusion coefficient assumption, $\Gamma_{fu} = \Gamma_{ox} = \rho D = \Gamma_\phi$, allows us to subtract equation (12.89) from s times equation (12.88) and combine the result into a single transport equation for ϕ:

$$\frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) = \operatorname{div}(\Gamma_\phi \operatorname{grad} \phi) + (s\dot{\omega}_{fu} - \dot{\omega}_{ox}) \tag{12.91}$$

From the one-step reaction assumption (12.87), we have $\dot{\omega}_{fu} = (1/s)\dot{\omega}_{ox}$, giving $(s\dot{\omega}_{fu} - \dot{\omega}_{ox}) = 0$, and equation (12.91) reduces to

$$\frac{\partial(\rho\phi)}{\partial t} + \operatorname{div}(\rho\phi\mathbf{u}) = \operatorname{div}(\Gamma_\phi \operatorname{grad} \phi) \tag{12.92}$$

Hence, ϕ is a passive scalar; it obeys the scalar transport equation without source terms. A non-dimensional variable ξ called the **mixture fraction** may be defined in terms of ϕ as follows:

$$\boxed{\xi = \frac{\phi - \phi_0}{\phi_1 - \phi_0}} \tag{12.93}$$

where suffix 0 denotes the oxidant stream and 1 denotes the fuel stream. The local value of ξ equals 0 if the mixture at a point contains only oxidant and equals 1 if it contains only fuel.

Equation (12.93) may be written in expanded form as

$$\boxed{\xi = \frac{[sY_{fu} - Y_{ox}] - [sY_{fu} - Y_{ox}]_0}{[sY_{fu} - Y_{ox}]_1 - [sY_{fu} - Y_{ox}]_0}} \tag{12.94}$$

If the *fuel stream* has *fuel only* we have

$$[Y_{fu}]_1 = 1 \quad [Y_{ox}]_1 = 0 \tag{12.95}$$

and if the *oxidant stream contains no fuel* we have

$$[Y_{fu}]_0 = 0 \quad [Y_{ox}]_0 = 1 \tag{12.96}$$

In such conditions equation (12.94) may be simplified as follows:

$$\boxed{\xi = \frac{[sY_{fu} - Y_{ox}] - [-Y_{ox}]_0}{[sY_{fu}]_1 - [-Y_{ox}]_0} = \frac{sY_{fu} - Y_{ox} + Y_{ox,0}}{sY_{fu,1} + Y_{ox,0}}} \tag{12.97}$$

In a stoichiometric mixture neither fuel nor oxygen is present in the products, and the **stoichiometric mixture fraction** ξ_{st} may be defined as

$$\boxed{\xi_{st} = \frac{Y_{ox,0}}{sY_{fu,1} + Y_{ox,0}}} \tag{12.98}$$

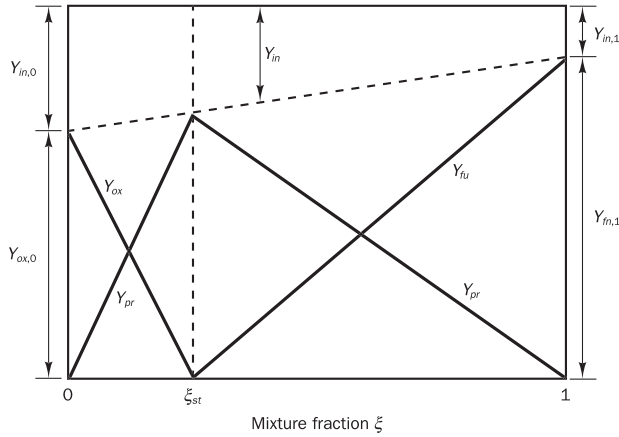
Fast chemistry implies that, at a certain location with a **lean mixture**, there is an excess of oxidant and no fuel is present in the products. Hence $Y_{fu} = 0$ if $Y_{ox} > 0$, so

$$\text{if } \xi < \xi_{st} \quad \text{then} \quad \xi = \frac{-Y_{ox} + Y_{ox,0}}{sY_{fu,1} + Y_{ox,0}} \tag{12.99}$$

Conversely, in a **rich mixture** there is a local excess of fuel in the mixture and there is no oxidant in the products. Hence $Y_{ox} = 0$ if $Y_{fu} > 0$, so

$$\text{if } \xi > \xi_{st} \quad \text{then} \quad \xi = \frac{sY_{fu} + Y_{ox,0}}{sY_{fu,1} + Y_{ox,0}} \tag{12.100}$$

The above formulae show that the mass fractions of the fuel Y_{fu} and oxygen Y_{ox} are linearly related to the mixture fraction ξ. This is illustrated graphically in Figure 12.2.

Figure 12.2 Mixing and fast reaction between fuel and oxidant streams (SCRC relationships)

By equation (12.93) ξ is linearly related to ϕ so the **mixture fraction** is also a passive scalar and obeys the **transport equation**

$$\boxed{\frac{\partial(\rho\xi)}{\partial t} + \text{div}(\rho\xi\mathbf{u}) = \text{div}(\Gamma_\xi \,\text{grad}\, \xi)} \tag{12.101}$$

Written in suffix notation the transport equation for the mixture fraction is

$$\frac{\partial}{\partial t}(\rho\xi) + \frac{\partial}{\partial x_i}(\rho u_i \xi) = \frac{\partial}{\partial x_i}\left(\Gamma_\xi \frac{\partial \xi}{\partial x_i}\right) \tag{12.102}$$

To obtain the distribution of ξ we solve equation (12.101) subject to suitable boundary conditions, e.g. mixture fractions of fuel and oxidant inlet streams are known, zero normal flux of ξ across solid walls and zero-gradient outflow conditions. Given the resulting mixture fraction we can rearrange

equations (12.98)–(12.100) to give values for oxygen and fuel mass fractions after combustion:

$$\xi_{st} \le \xi < 1: \quad Y_{ox} = 0 \quad Y_{fu} = \frac{\xi - \xi_{st}}{1 - \xi_{st}} Y_{fu,1} \tag{12.103}$$

$$0 < \xi < \xi_{st}: \quad Y_{fu} = 0 \quad Y_{ox} = \frac{\xi_{st} - \xi}{\xi_{st}} Y_{ox,0} \tag{12.104}$$

The reactants may be accompanied by inert species such as N_2, which do not take part in the reaction. The mass fraction of inert species in the mixture varies linearly with ξ, as illustrated in Figure 12.2. Simple geometry gives the total mass fraction of the inert species Y_{in} after combustion at any value of ξ as

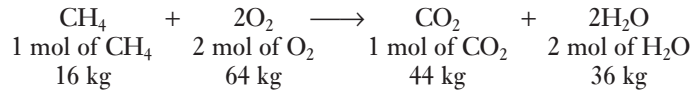
$$Y_{in} = Y_{in,0}(1 - \xi) + Y_{in,1}\xi \tag{12.105}$$

The mass fraction of products (Y_{pr}) of combustion may be obtained from

$$Y_{pr} = 1 - (Y_{fu} + Y_{ox} + Y_{in}) \tag{12.106}$$

The above equations, (12.101) and (12.103)–(12.106), represent the SCRS model.

When the reaction products contain two or more species, the ratio of the mass fraction of each component to the total product mass fraction is known from the equation for the chemical reaction and can be used to deduce the mass fraction of different product components. For example, consider the burning of methane with O_2:

CH_4	+	$2O_2$	$\longrightarrow$	CO_2	+	$2H_2O$
1 mol of CH_4		2 mol of O_2		1 mol of CO_2		2 mol of H_2O
16 kg		64 kg		44 kg		36 kg

Ratio of CO_2 in products by mass (r_{CO_2}) = 44/80
Ratio of H_2O in products by mass (r_{H_2O}) = 36/80

If the product mass fraction from equation (12.106) is Y_{pr} then the CO_2 mass fraction in the products is $Y_{pr}r_{CO_2}$ and the H_2O mass fraction is $Y_{pr}r_{H_2O}$.

The SCRS model has made the following simplifications: (i) single-step reaction between fuel and oxidant, and (ii) one reactant which is locally in excess causes *all* the other reactant to be consumed stoichiometrically to form reaction products. These assumptions fix algebraic relationships between the mixture fraction ξ and all the mass fractions Y_{fu}, Y_{ox}, Y_{in} and Y_{pr}. As a consequence of the additional assumption that the mass diffusion coefficients of all species are equal, it is only necessary to solve *one* partial differential equation for ξ to calculate combusting flows rather than individual partial differential equations for the mass fraction of each species. An example which uses this approach for combustion calculations is presented below.

12.16 Modelling of a laminar diffusion flame – an example

The SCRS model can be readily applied to calculate temperatures and species distribution in laminar diffusion flames. For example, consider the axisymmetric laminar non-premixed diffusion flame geometry shown in

Figure 12.3 Schematic of the problem considered

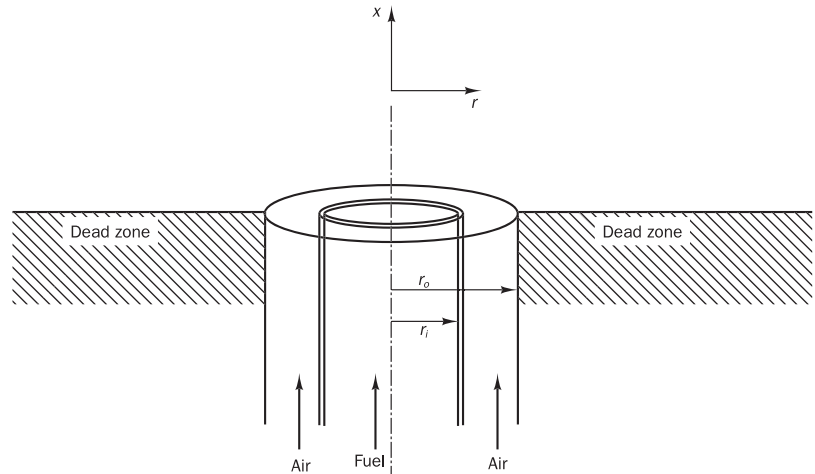
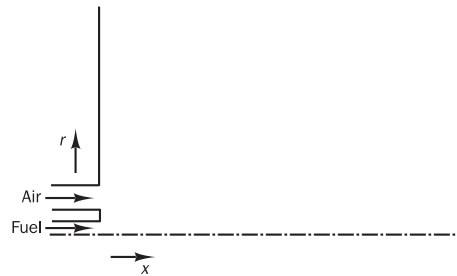



Figure 12.3. The geometry considered here is an experimental configuration which has been used by a number of studies documented in the combustion literature (Bennett and Smooke, 1998; Smooke *et al.*, 1990; Smooke, 1991). For illustrative purposes we use our own operating conditions to formulate the problem in this example. The radius of the fuel jet (r_i) is 0.2 cm and the radius (r_o) of the co-flowing air jet is 2.5 cm. The thickness of the wall between fuel and air streams is 0.05 cm. Let us consider burning of a pure methane jet with co-flowing air. Both fuel and air velocities are taken as 0.2 m/s (20 cm/s) and enter at a temperature of 25°C (298 K). We would like to calculate the resulting flame temperature and major species distribution field for this flame.

For calculation convenience we turn the actual geometry shown in Figure 12.3 through 90°. The axisymmetry of the problem allows us to adopt a cylindrical x,r coordinate system to formulate the problem as shown in Figure 12.4. Here we use u to represent the velocity component in the (axial) x-direction and v to represent the velocity component in the (radial) r-direction.

Figure 12.4 A part of the computational geometry

The governing equations for steady laminar flow in cylindrical coordinates in expanded form are as follows (see section 2.3 and section 12.14 above).

Continuity

$$\frac{\partial(\rho u)}{\partial x} + \frac{1}{r}\frac{\partial(\rho r v)}{\partial r} = 0 \tag{12.107}$$

Momentum equations

u-momentum equation:

$$\frac{\partial}{\partial x}(\rho ruu) + \frac{\partial}{\partial r}(\rho ruv) = \frac{\partial}{\partial x}(r\tau_{xx}) + \frac{\partial}{\partial r}(r\tau_{rx}) - r\frac{\partial p}{\partial x} \tag{12.108}$$

v-momentum equation

$$\frac{\partial}{\partial x}(\rho ruv) + \frac{\partial}{\partial r}(\rho rvv) = \frac{\partial}{\partial x}(r\tau_{rx}) + \frac{\partial}{\partial r}(r\tau_{rr}) - r\frac{\partial p}{\partial r} \tag{12.109}$$

The shear stress terms are

$$\tau_{xx} = \mu\left[2\frac{\partial u}{\partial x} - \frac{2}{3}(\text{div } \mathbf{u})\right] \tag{12.110}$$

$$\tau_{rr} = \mu\left[2\frac{\partial v}{\partial r} - \frac{2}{3}(\text{div } \mathbf{u})\right] \tag{12.111}$$

$$\tau_{rx} = \mu\left[2\frac{\partial u}{\partial r} + \frac{\partial v}{\partial x}\right] \tag{12.112}$$

$$\text{div } \mathbf{u} = \frac{\partial u}{\partial x} + \frac{1}{r}\frac{\partial}{\partial r}(rv) \tag{12.113}$$

The momentum equations can be rearranged in the usual form as follows:

u-momentum equation:

$$\frac{\partial}{\partial x}(\rho ruu) + \frac{\partial}{\partial r}(\rho ruv) = \frac{\partial}{\partial x}\left(r\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\mu\frac{\partial u}{\partial r}\right) - r\frac{\partial p}{\partial x} + S_u \tag{12.114}$$

v-momentum equation:

$$\frac{\partial}{\partial x}(\rho ruv) + \frac{\partial}{\partial r}(\rho rvv) = \frac{\partial}{\partial x}\left(r\mu\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\mu\frac{\partial v}{\partial r}\right) - r\frac{\partial p}{\partial r} + S_v \tag{12.115}$$

where S_u and S_v contain the additional terms arising from the shear stress terms.

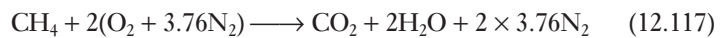
Enthalpy equation

$$\frac{\partial}{\partial x}(r\rho uh) + \frac{\partial}{\partial r}(r\rho vh) = \frac{\partial}{\partial x}\left(rD_h\frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial r}\left(rD_h\frac{\partial h}{\partial r}\right) + S_{rad} \tag{12.116}$$

where D_h is the diffusion coefficient for enthalpy (i.e. $D_h \equiv \alpha = k/\rho C_p$) and S_{rad} is the radiation source (or sink) term.

Combustion model

Here we use the SCRC model described in section 12.15. The assumption of fast chemistry gives the stoichiometric equation

$$CH_4 + 2(O_2 + 3.76N_2) \longrightarrow CO_2 + 2H_2O + 2 \times 3.76N_2 \tag{12.117}$$

The stoichiometric oxygen/fuel ratio by mass is

$$s = 2 \times (MW)_{O_2}/(MW)_{CH_4} = 2 \times 32/16 = 4$$

The mass fraction of fuel in the fuel stream is

$$Y_{fu,1} = 1.0$$

The mass fraction of oxygen in the air stream is

$$Y_{ox,0} = 0.233 \quad \text{(see section 12.5)}$$

The mass fraction of inert (N_2) in the air stream is

$$Y_{in,0} = 0.767$$

We solve the following equation for the mixture fraction:

$$\frac{\partial}{\partial x}(r\rho u\xi) + \frac{\partial}{\partial r}(r\rho v\xi) = \frac{\partial}{\partial x}\left(r\Gamma_\xi \frac{\partial \xi}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\Gamma_\xi \frac{\partial \xi}{\partial r}\right) \tag{12.118}$$

The mass fractions for the fuel and oxidant streams are as follows.

Since the fuel stream has fuel only we have

$$[Y_{fu}]_1 = 1 \quad [Y_{ox}]_1 = 0 \tag{12.119}$$

The oxidant stream contains no fuel but oxygen and nitrogen (inert), so we have

$$[Y_{fu}]_0 = 0 \quad [Y_{ox}]_0 = 0.233 \quad [Y_{in}]_0 = 0.767 \tag{12.120}$$

The stoichiometric mixture fraction from equation (12.98) is

$$\xi_{st} = \frac{0.233}{4 \times 1 + 0.233} = 0.055 \tag{12.121}$$

If heat loss by radiation is considered in the calculation then solution of the enthalpy equation is required. If radiation is considered to be negligible then it can be seen that from equation (12.116) when $S_{rad} = 0$ the transport equation for enthalpy becomes another conserved scalar equation like the mixture fraction equation. Therefore the enthalpy and mixture fraction are both scalars and linearly related. The relationship can be further illustrated as below.

Taking the reference temperature as zero, enthalpy is defined as

$$h = Y_{fu}\Delta h_{fuel} + c_p T \tag{12.122}$$

Enthalpy of the fuel stream where $\xi = 1$ is

$$h_{fu,in} = Y_{fu,in} \times \Delta h_{fuel} + Y_{fu} c_p \times T_{fu,in}$$

where Δh_{fuel} is the enthalpy or formation of fuel. Enthalpy of the air stream where $\xi = 0$ is

$$h_{air,in} = c_p \times T_{air,in}$$

If we define a non-dimensional enthalpy as

$$h^* = \frac{h - h_{air,in}}{h_{fu,in} - h_{air,in}} \tag{12.123}$$

we can see that when $\xi = 0$, $Y_{fu} = 0$, $h^* = 0$ and when $\xi = 1$, $Y_{fu} = 1$, $h^* = 1$. If we make the usual simplifying assumptions – single diffusion coefficient, unity

Lewis number, negligible pressure work and radiation source – for enthalpy transport, the governing transport equations for the enthalpy and mixture fraction are the same (both variables are passive scalars). The resulting spatial distributions of non-dimensional enthalpy h^* and mixture fraction will be the same (it is easy to verify that the boundary conditions for both variables are also identical in this problem). Therefore, we do not need to solve a separate transport equation for enthalpy, but can calculate this variable from

$$\xi = h^* = \frac{h - h_{air,in}}{h_{fu,in} - h_{air,in}} \tag{12.124}$$

Once the enthalpy value and mass fraction of fuel Y_{fu} are known from SCRC relationships, the temperature is obtained from

$$T = \frac{h - Y_{fu}(\Delta h_{fu})}{\bar{c}_p} \tag{12.125}$$

The boundary conditions for the problem are: at the inlet, $u_{in} = 0.2$ m/s for both fuel and air streams; $\xi_{fu} = 1.0$ for the fuel stream; and $\xi_{air} = 0.0$ for the air stream. Zero-velocity and zero-mixture-fraction-gradient wall boundary conditions are used at all solid walls and a constant pressure boundary condition is imposed on open boundaries with pressure set to ambient. At the symmetry axis all gradients are set to zero.

Solution of the fluid flow equations and the mixture fraction equation gives the distribution of mixture fraction which defines the flame structure and species distribution. Equations (12.103) and (12.104) are used to obtain species mass fractions. Equations (12.124) and (12.125) give the enthalpy and temperature fields. Using pressure and temperature the density field is obtained from (12.86). Because of the coupled nature of the equations, the entire solution process is iterative. Figure 12.5 shows some typical results obtained from this simulation. To highlight the consequences of the fast chemistry assumption of the SCRS model we show radial profiles of temperature and species mass fraction at three different points along the axis as well as the temperature contours in Figure 12.5. The stoichiometric contour corresponds with the contour of maximum temperature and defines the flame. In the region inside the stoichiometric contour $\xi > \xi_{st}$, so by condition (12.103) fuel exists here without oxidant. On the other hand, outside the stoichiometric contour $\xi < \xi_{st}$, so by (12.104) no fuel can exist. At the axial location (a) where $\xi > \xi_{st}$ the radial profiles show no oxygen near the axis, and fuel which gets completely consumed at the ξ_{st} contour. At axial location (b) where centreline $\xi = \xi_{st}$ neither fuel nor oxygen is present and the temperature is at a maximum. At axial location (c) where centreline $\xi < \xi_{st}$ no fuel exists and the temperature is lower.

Calculations with finite rate chemistry

To include finite rate and detailed chemistry in this combustion calculation one has to consider a detailed mechanism and solve many species transport equations of the form

$$\frac{\partial}{\partial x}(r\rho u Y_k) + \frac{\partial}{\partial x}(r\rho v Y_k) = \frac{\partial}{\partial x}\left(r\rho D_k \frac{\partial Y_k}{\partial x}\right) + \frac{\partial}{\partial r}\left(r\rho D_k \frac{\partial Y_k}{\partial r}\right) + r\dot{\omega}_k \tag{12.126}$$

where $\dot{\omega}_k$ is the rate of generation of species k, which is determined from chemical kinetic expressions such as (12.69)–(12.71). Numerical solutions

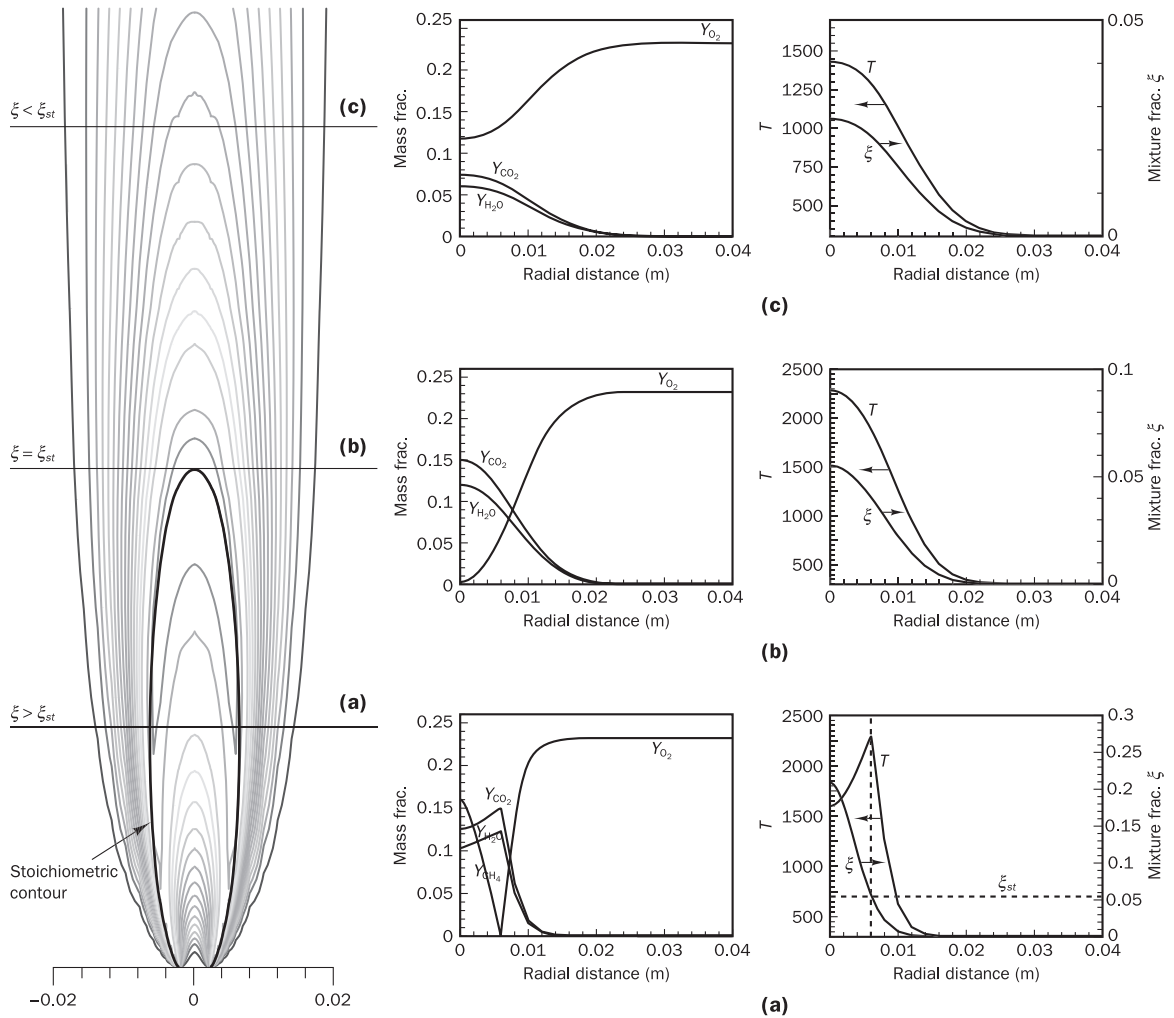



Figure 12.5 Calculated flame structure of a laminar diffusion flame

including detailed and finite rate chemistry result in some variations to the curves shown in Figure 12.5. The main difference is an overlap of fuel and oxygen profiles and small curvature around the stoichiometric mixture fraction showing a corresponding drop in temperature: see Warnatz *et al*. (2001). Comprehensive combustion calculations for this geometry (for different operating conditions) including detailed chemistry can be found in Smooke *et al*. (1990), Bennett and Smooke (1998), Smooke and Bennett (2001).

The method presented above illustrates how CFD can be used for laminar non-premixed combustion calculations. Unfortunately, the fast chemistry assumption does not give adequate details of minor species, since it is necessary to include more detailed chemistry and finite rate kinetics for a more comprehensive account of combustion. Prediction of pollutants, such as NO_x discussed later in this chapter, inevitably requires inclusion of transport equations for the most important minor species.

12.17 CFD calculation of turbulent non-premixed combustion

The CFD calculation of turbulent non-premixed combustion is not as straightforward as laminar calculations, even with the fast chemistry assumption. In Chapter 3 we showed that Reynolds averaging and modelling of the resulting averages of fluctuating product terms made it possible to predict incompressible turbulent flow fields. Equations governing turbulent non-premixed combustion also require averaging and modelling. The first problem that needs to be addressed is the fact that strong and highly localised heat generation in combusting flows causes the density to vary as a function of position in combusting flows. There will also be density fluctuations if the flow is turbulent. The Reynolds decomposition of a general flow variable is as follows:

$$\phi = \bar{\phi} + \phi'$$

For the variables in a reacting flow this yields

$$u_i = \bar{u}_i + u_i'$$
$$p = \bar{p} + p'$$
$$\rho = \bar{\rho} + \rho'$$
$$h = \bar{h} + h'$$
$$T = \bar{T} + T'$$
$$Y_k = \bar{Y}_k + Y_k'$$

It is easy to demonstrate that the presence of density fluctuations gives rise to additional terms when Reynolds averaging is used. For example, consider the instantaneous continuity equation in suffix notation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u_i)}{\partial x_i} = 0 \tag{12.127}$$

After substituting for u and ρ the Reynolds-averaged equation is

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial(\bar{\rho}\bar{u}_i)}{\partial x_i} + \frac{\partial(\overline{\rho' u_i'})}{\partial x_i} = 0 \tag{12.128}$$

Compare this with the Reynolds-averaged equation for a constant density flow:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho \bar{u}_i) = 0 \tag{12.129}$$

The additional term $\partial(\overline{\rho' u_i'})/\partial x_i$ in equation (12.128) arises from correlations between the velocity and density fluctuations in a reacting flow and has to be modelled. Many more terms of this type appear in the Reynolds-averaged momentum, scalar and species transport equations.

To reduce the number of separate terms requiring modelling in reacting flows with variable density, we use a density-weighted averaging procedure known as *Favre* averaging (Favre, 1969; Jones and Whitelaw, 1982).

In Favre averaging the density-weighted mean velocity is defined as follows:

$$\tilde{u} = \frac{\overline{\rho u}}{\bar{\rho}} \tag{12.130}$$

The instantaneous velocity u is written as

$$u = \tilde{u} + u'' = \frac{\overline{\rho u}}{\bar{\rho}} + u'' \tag{12.131}$$

In contrast to the Reynolds decomposition, where u' represents a turbulent velocity fluctuation, the quantity u'' also includes effects of density fluctuations. If the flow is incompressible, the density is constant, so $\tilde{u} = \bar{u}$ and $u'' \equiv u'$.

In the convective term of the continuity equation we require the Favre average of the product ρu_i. Multiplying equation (12.131) by ρ we obtain

$$\rho u = \rho(\tilde{u} + u'') = \rho\tilde{u} + \rho u'' \tag{12.132}$$

Time averaging equation (12.132) we get

$$\overline{\rho u} = \bar{\rho}\tilde{u} + \overline{\rho u''} \tag{12.133}$$

By definition of Favre averaging (12.130) and equation (12.131) we have $\overline{\rho u''} = 0$. Now the Favre-averaged continuity equation can be obtained:

$$\boxed{\frac{\partial\bar{\rho}}{\partial t} + \frac{\partial(\bar{\rho}\tilde{u}_i)}{\partial x_i} = 0} \tag{12.134}$$

Unlike Reynolds-averaged continuity equation (12.128), this equation has the same form as the original continuity equation (12.127) and Reynolds-averaged *constant density* equation (12.129), except that the mean velocity is the density-weighted Favre-averaged velocity.

The Favre-averaging procedure considerably reduces the number of additional terms in the other flow equations arising from density fluctuations. For example, Reynolds averaging of the convective term $\rho u_i u_j$ in the momentum equation gives

$$\begin{aligned}\overline{\rho u_i u_j} &= \overline{(\rho + \rho')(\bar{u}_i + u_i')(\bar{u}_j + u_j')} \\ &= \bar{\rho}\bar{u}_i\bar{u}_j + \bar{\rho}\overline{u'v'} + \bar{u}_i\overline{\rho' u_j'} + \bar{u}_j\overline{\rho' u_i'} + \overline{\rho' u_i' u_j'}\end{aligned} \tag{12.135}$$

Favre averaging of the same term leads to

$$\begin{aligned}\overline{\rho u_i u_j} &= \overline{\rho(\tilde{u}_i + u_i'')(\tilde{u}_j + u_j'')} \\ &= \bar{\rho}\tilde{u}_i\tilde{u}_j + \overline{\rho u_i'' u_j''}\end{aligned} \tag{12.136}$$

This clearly highlights the reduced number of unknown correlations (products of fluctuating quantities) which is the main advantage of Favre averaging. However, care must be taken when comparing results obtained by solving Favre-averaged equations with experimental data which are often time averaged. Therefore, conversion of Favre-averaged quantities to time-averaged quantities is necessary. In order to do this we need to know more about the turbulent fluctuations first. In section 12.19 we look at descriptions of turbulent fluctuations in terms of probability density functions and address the conversion problem.

Favre averaging of other governing equations, i.e. momentum, energy, scalar transport and species transport, yields the same form of equations as those for turbulent constant density flows. Without presenting the derivations we give the set of Favre-averaged equations used to model turbulent combusting flows as follows:

Continuity

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial}{\partial x_i} \bar{\rho} \tilde{u}_i = 0 \quad (12.137)$$

Momentum

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{u}_i) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_i\tilde{u}_j) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(\bar{\tau}_{ij} - \overline{\rho u_i'' u_j''}) \quad (12.138)$$

Enthalpy

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{h}) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{h}) = \frac{\partial}{\partial x_j}\left(\Gamma_h \frac{\partial \tilde{h}}{\partial x_j}\right) + \bar{S}_h \quad (12.139)$$

where $\Gamma_h = (\mu/\sigma + \mu_t/\sigma_h)$, σ_h = turbulent Prandtl number

Mixture fraction in conserved scalar transport model for combustion

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{\xi}) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{\xi}) = \frac{\partial}{\partial x_j}\left(\Gamma_\xi \frac{\partial \tilde{\xi}}{\partial x_j}\right) \quad (12.140)$$

where $\Gamma_\xi = (\mu/\sigma + \mu_t/\sigma_\xi)$, σ_ξ = turbulent Schmidt number

In the two equations (12.139) and (12.140) the familiar gradient diffusion treatment has been used to model averages of products of fluctuating quantities, which we first encountered in section 3.7, where it was used to model the diffusion of turbulence quantities k and ε.

Species conservation (used in more detailed combustion models) is given by

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{Y}_k) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{Y}_k) = \frac{\partial}{\partial x_j}\left(\bar{\rho}D_k \frac{\partial \tilde{Y}_k}{\partial x_j} - \overline{\rho Y_k'' u_j''}\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}D_k \overline{\frac{\partial Y_k''}{\partial x_j}}\right) + \tilde{\dot{\omega}}_k \quad (12.141)$$

where $\tilde{\dot{\omega}}_k$ is the Favre-averaged reaction rate.

Again applying gradient diffusion assumption gives

$$\overline{\rho Y_k'' u_i''} = \frac{\mu_t}{\sigma_k}\frac{\partial \tilde{Y}_k}{\partial x_j}$$

where σ_k is the turbulent Schmidt number for species k, and like other equations we may write the transport equation for species as

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{Y}_k) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{Y}_k) = \frac{\partial}{\partial x_j}\left(\Gamma_k \frac{\partial \tilde{Y}_k}{\partial x_j}\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}D_k \overline{\frac{\partial Y_k''}{\partial x_j}}\right) + \tilde{\dot{\omega}}_k \quad (12.142)$$

where $\Gamma_k = (\mu/\sigma + \mu_t/\sigma_k)$. Note here that subscript k stands for species and should not be confused with the similar effective diffusion coefficient used in the turbulent kinetic energy equation.

It should be noted that the Favre-averaged forms of the continuity, enthalpy and mixture fraction equations do not include terms involving density fluctuations. The Reynolds stress terms in the momentum equations are modelled using the same turbulence modelling approach used for time-averaged equations. For example, the k–ε model may be applied with Favre-averaged transport equations for k and ε. The species equation has been simplified as much as possible, but the involvement of species and density in the reaction rate term still poses a problem.

The main problem in turbulent combusting flows arises from the averaging of the species generation term $\bar{\dot{\omega}}_k$. In section 12.11 it was shown that for a simple single-step reaction of the form

$$\text{Fuel} + s\ \text{Oxident} \rightarrow (1+s)\text{Products} \tag{12.143}$$

$$\dot{\omega}_{fu} = AT^{\alpha} C_{fu} C_{ox}^{s} \exp\left(\frac{-E_a}{R_u T}\right) \tag{12.144}$$

For the simplest case where $s = 1$, and after converting concentrations into mass fractions (see (12.73)), the generation or consumption of the species fuel may be written as

$$\dot{\omega}_{fu} = A'\rho^2 T^{\alpha} Y_{fu} Y_{ox} \exp\left(\frac{-E_a}{R_u T}\right) \tag{12.145}$$

where 'fuel' and 'oxidant' are the reactants, and A' is an appropriately modified constant. This expression can be written in a slightly different form as

$$\dot{\omega}_{fu} = A'\rho^2 T^{\alpha} Y_{fu} Y_{ox} \exp\left(\frac{-T_A}{T}\right) \tag{12.146}$$

where $T_A = E_a/R_u$ is called the activation temperature and ρ is density. As the reaction rate is highly non-linear, the average reaction rate $\bar{\dot{\omega}}_{fu}$ cannot be easily expressed as a function of Favre-averaged mass fractions $\tilde{Y}_{fu}$, $\tilde{Y}_{ox}$, the mean density $\bar{\rho}$ and mean temperature $\tilde{T}$. A Taylor series expansion may be used for the exponential term as an attempt to obtain an averaged expression for $\bar{\dot{\omega}}_{fu}$. Using such an approach, the averaged reaction rate may be expressed as (Veynante and Vervisch, 2002)

$$\bar{\dot{\omega}}_{fu} = -A\bar{\rho}^2 \tilde{T}^{\alpha} \tilde{Y}_{fu} \tilde{Y}_{ox} \exp\left(\frac{-T_A}{\tilde{T}}\right) \times \left[1 + \frac{\widetilde{Y''_{fu} Y''_{ox}}}{\tilde{Y}_{fu}\tilde{Y}_{ox}} + (P_1 + Q_1)\left(\frac{\widetilde{Y''_{fu} T''}}{\tilde{Y}_{fu}\tilde{T}} + \frac{\widetilde{Y''_{ox} T''}}{\tilde{Y}_{ox}\tilde{T}}\right) + (P_2 + Q_2 + P_1 Q_1)\left(\frac{\widetilde{Y''_{fu} T''^2}}{\tilde{Y}_{fu}\tilde{T}^2} + \frac{\widetilde{Y''_{ox} T''^2}}{\tilde{Y}_{ox}\tilde{T}^2}\right) + \ldots\right] \tag{12.147}$$

In this expression P_1, Q_1, P_2, Q_2 etc. are terms of a series defined in Veynante and Vervisch (2002). We do not intend to describe the details here; the expression is shown just to highlight the complexity of the attempt to find an

average. It can be seen that the averaged $\bar{\omega}_{fu}$ term in this way involves correlations such as $Y''_{fu}T''$, $Y''_{ox}T''$, $Y''_{fu}T''^2$ etc. and many others that are unknown and have to be modelled. The above expression is for a single-step reaction. When realistic chemical schemes are introduced involving many reactions and species, it is not possible to model these correlations. Therefore, a considerable amount of effort in turbulent combustion modelling has been directed towards the development of models which avoid the solution of averaged species mass fraction equations (12.142).

12.18 SCRS model for turbulent combustion

Given the difficulties associated with modelling of the source term in the Favre-averaged species transport equation (12.142) it is useful to examine the effect of turbulent fluctuations on the SCRS model. It was shown earlier how the fast chemical reaction using a single-step reaction could be used to develop a transport equation for the mixture fraction. In turbulent flows the mixture fraction equation (12.102) applies in Favre-averaged form (12.140). A Favre-averaged equation for enthalpy is also solved in cases where radiation effects and other heat losses are significant. However, the calculation of mean species and temperature using the field values of $\tilde{\xi}$ and $\tilde{h}$ is not as straightforward as in the laminar case. The linear relationships between species mass fractions and mixture fraction relate to instantaneous mixture fraction ξ and not to $\tilde{\xi}$. The same applies to the temperature–enthalpy relationship. To calculate mean values of $\tilde{Y}_i$ and $\tilde{T}$ we need to know the statistics of the variables (T, Y_i, ρ) as a function of ξ. This is where an approach known as the probability density function method is used in turbulent combustion calculations.

12.19 Probability density function approach

We introduce a statistical approach to calculate mean quantities using the probability density function (pdf) for fluctuating scalars (in this case the mixture fraction). Mathematical descriptions of probability density functions can be found in popular mathematics textbooks (see e.g. Evans *et al*., 2000) and we do not attempt to describe the fundamentals here. The reader is also referred to Kuo (1986) for further details and applications to the theory of turbulent combustion.

For a random variable ϕ, the probability function $F_\phi(\psi)$ is defined as

$$F_\phi(\psi) = \text{Prob}\{\phi < \psi\} \tag{12.148}$$

where Prob$\{\phi < \psi\}$ is the probability that ϕ is smaller than given value ψ.

Then the **probability density function** $P_\phi(\psi)$ is defined as the derivative of the distribution function $F_\phi(\psi)$:

$$P_\phi(\psi) = \frac{\mathrm{d}F_\phi(\psi)}{\mathrm{d}\psi} \tag{12.149}$$

The properties of a probability density function or pdf are

(i) $P_\phi(\psi) \geq 0$

(ii) $\displaystyle\int_{-\infty}^{\infty} P_\phi(\psi)\,\mathrm{d}\psi = 1$

The probability density function $P(\psi)$ can be used to calculate the **mean of any property** $q(\psi)$ which also depends on ψ in the following manner (see Kuo, 1986):

$$\bar{q} = \int_{-\infty}^{\infty} q(\psi)P(\psi)\mathrm{d}\psi \tag{12.150}$$

In the context of combustion calculations we need to define the density-weighted probability density function $\tilde{P}(\xi)$ for the mixture fraction ξ at every location. The density-weighted pdf $\tilde{P}(\xi)$ and unweighted pdf $P(\xi)$ are related as follows:

$$\tilde{P}(\xi) = \frac{\rho(\xi)}{\bar{\rho}}P(\xi) \tag{12.151}$$

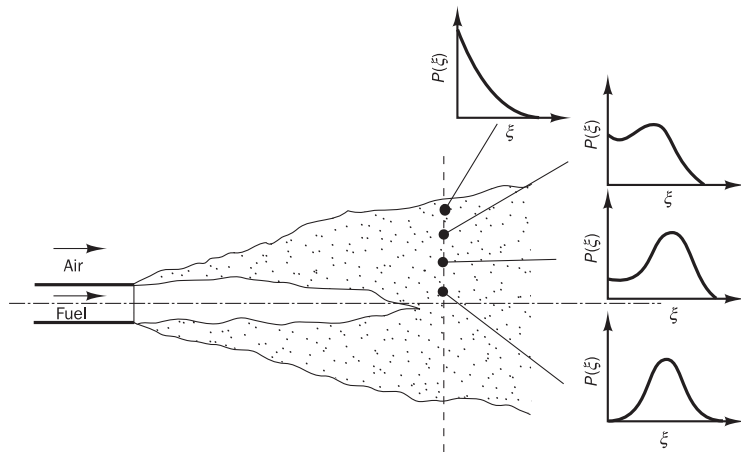
According to Jones and Whitelaw (1982), the density-weighted average of any scalar quantity ϕ, which is itself a function of ξ, may now be obtained from

$$\tilde{\phi} = \int_{0}^{1} \phi(\xi)\tilde{P}(\xi)\,\mathrm{d}\xi \tag{12.152}$$

For example, the density-weighted average of the mass fraction of species i is given by

$$\tilde{Y}_k = \int_{0}^{1} Y_k(\xi)\tilde{P}(\xi)\,\mathrm{d}\xi \tag{12.153}$$

In the SCRC model the relationship between Y_k and ξ, i.e. $Y_k = Y_k(\xi)$, is known from the fast chemistry assumptions (equations (12.103)–(12.106)) or in graphical form as shown in Figure 12.2. To obtain the density-weighted average $\tilde{Y}_k$ it is necessary to perform the integration (12.153), for which we need to know the pdf $\tilde{P}(\xi)$ at all locations. The form of the pdf varies for different types of flows. Figure 12.6 (see Bilger, 1980) shows observed pdfs at

Figure 12.6 Schematic forms of pdfs in a turbulent jet flame

various locations in a jet flame. Different analytical pdfs have been used to approximate measured distributions, but the clipped Gaussian and beta functions have been most successful. The interested reader is referred to Bilger and Kent (1974), Lockwood and Naguib (1975), Bilger (1976), Pope (1976), Lockwood and Monib (1980) and Pope (1985), among others, for further details. The beta probability function for $\tilde{P}(\xi)$ is currently the most popular among modellers and has been incorporated in combustion modelling procedures of all the leading commercial CFD codes. Some details of the beta function approach are given below.

A further point to note is that problems arise when CFD results from Favre-averaged flow equations are compared with time-averaged experimental data. Suppose, however, that we know the pdf $\tilde{P}(\xi)$ and the relationships $\phi(\xi)$ and $\rho(\xi)$. Now we can replace u by ϕ/ρ in definition (12.130). This enables us to compute the time-averaged mean of scalar $\bar{\phi}$ and time-averaged density $\bar{\rho}$ as follows:

$$\bar{\phi} = \bar{\rho} \int_0^1 \frac{\phi(\xi)}{\rho(\xi)} \tilde{P}(\xi) d\xi \tag{12.154}$$

and, using $\phi = 1$ in (12.154),

$$\bar{\rho} = \left[\int_0^1 \frac{\tilde{P}(\xi)}{\rho(\xi)} d\xi \right]^{-1} \tag{12.155}$$

12.20 Beta pdf

The beta pdf (β-pdf) is defined as

$$\boxed{\tilde{P}(\xi) = \frac{\xi^{\alpha-1}(1-\xi)^{\beta-1}\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}} \tag{12.156}$$

Here $\Gamma(z)$ denotes the gamma function (see Evans *et al*., 2000; Abramowitz and Stegun, 1970). The shape of the β-pdf depends on the values of parameters α and β, which must both be positive ($\alpha > 0$ and $\beta > 0$). This is illustrated in Figure 12.7, which shows that it is possible to vary α and β to give a reasonable match with measured pdfs such as those in Figure 12.6.

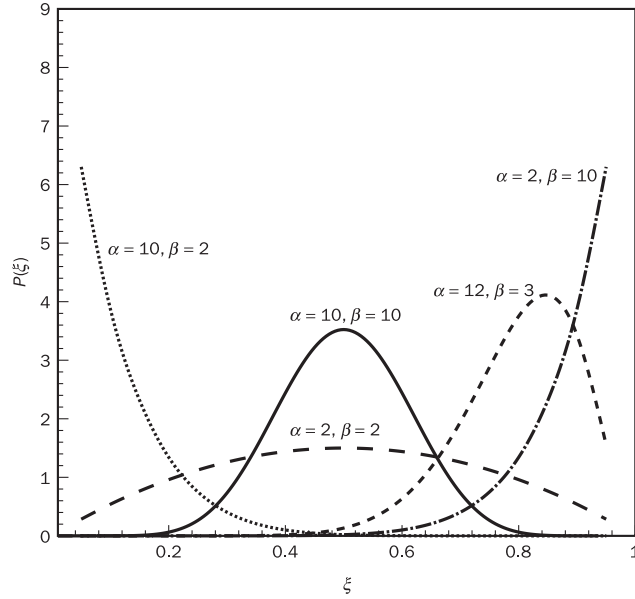
It can be shown that α and β can be determined from the mean and variance of ξ as follows:

$$\alpha = \tilde{\xi}\left[\frac{\tilde{\xi}(1-\tilde{\xi})}{\widetilde{\xi''^2}} - 1\right]$$

$$\beta = (1-\tilde{\xi})\frac{\alpha}{\tilde{\xi}} \tag{12.157}$$

where $\tilde{\xi}$ is the Favre mean of ξ and $\widetilde{\xi''^2}$ is the Favre-averaged variance of ξ.

DNS calculations of pdfs in turbulent combusting flows also support the use of a β-pdf: see for example Swaminathan and Bilger (1999). The reader is also referred to Warnatz *et al*. (2001), Libby and Williams (1994), Lentini

Figure 12.7 Behaviour of the beta function for different values of α and β

(1994) and Liu *et al.* (2002) for further details on pdf-based combustion calculations.

To generate the β-pdf using (12.156) and (12.157) it is necessary to solve the Favre-averaged equation for mean mixture fraction $\tilde{\xi}$ (12.140) as well as a further transport equation for mixture fraction variance $\tilde{\xi''^2}$. Without derivation we write the modelled transport equation for $\tilde{\xi''^2}$ in suffix notation (see Lockwood, 1977):

$$\boxed{\frac{\partial}{\partial t}(\bar{\rho}\tilde{\xi''^2}) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{\xi''^2}) = \frac{\partial}{\partial x_j}\left(\Gamma_\xi \frac{\partial \tilde{\xi''^2}}{\partial x_j}\right) + C_{g1}\mu_t\left(\frac{\partial \tilde{\xi''^2}}{\partial x_j}\right)^2 - C_{g2}\frac{\tilde{\varepsilon}}{\tilde{k}}\rho\tilde{\xi''^2}} \quad (12.158)$$

On the left hand side of this equation we have the transient and convective terms. All terms on the right hand side are modelled. The first term represents turbulent diffusion of mixture fraction variance along its gradients. We note that this model takes the diffusion coefficient for mixture fraction variance to be the same as the diffusion coefficient for mixture fraction. The second and third terms are the source and sink terms, respectively, where C_{g1} and C_{g2} are dimensionless model constants with values of 2.0 and 2.8 respectively.

Combining the β-pdf with the definition (12.152) of the density-weighted mean $\tilde{\phi}$ of any scalar flow variable ϕ we may now write

$$\tilde{\phi} = \int_0^1 \phi(\xi)\tilde{P}(\xi)\,\mathrm{d}\xi = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\int_0^1 \phi(\xi)\,\xi^{\alpha-1}(1-\xi)^{\beta-1}\mathrm{d}\xi \qquad (12.159)$$

If the relationship between variable ϕ and the *instantaneous* mixture fraction ξ is somehow known (e.g. Figure 12.2 for species mass fractions),

equation (12.159) can be numerically integrated for specified values of α and β. We normally use Romberg's method with the midpoint approximation (Press *et al.*, 1993). At the end points of the integration interval ($\xi = 0$, 1) the integrand becomes singular when the parameters α and β are less than 1.0. This singularity can be eliminated analytically using the method suggested by Bray *et al.* (1994) and Chen *et al.* (1994), which involves approximation of the integration as follows:

$$\tilde{\phi} = \int_0^1 \phi(\xi)\tilde{P}(\xi)\,\mathrm{d}\xi \cong \frac{\eta^\alpha}{\alpha}\phi(0) + \int_\eta^{1-\eta} \phi(\xi)\,\xi^{\alpha-1}(1-\xi)^{\beta-1}\mathrm{d}\xi + \frac{\eta^\beta}{\alpha}\phi(1) \tag{12.160}$$

where η is a very small number (say 10^{-30}).

Another numerical difficulty is that computed values of α and β in equations (12.157) may approach magnitudes of several hundred thousand in the iteration process (Chen *et al.*, 1994). This problem leads to overflow in the calculation of $\tilde{P}(\xi)$. According to the characteristics of the beta function, $\tilde{P}(\xi)$ is close to a delta function when either values of exponents α and β are sufficiently large. To avoid the overflow when α and β are large (say above 500) we approximate $\tilde{P}(\xi)$ by a delta function: $\tilde{P}(\xi) = \delta(\xi - \tilde{\xi})$. The density-weighted mean $\tilde{\phi}$ of scalar flow variable ϕ is now given by

$$\tilde{\phi} = \int_0^1 \phi(\xi)\tilde{P}(\xi)\,\mathrm{d}\xi = \int_0^1 \phi(\xi)\,\delta(\xi - \tilde{\xi})\,\mathrm{d}\xi = \phi(\tilde{\xi}) \tag{12.161}$$

Since the pdf is an assumed one – the β-pdf – this type of calculation is sometimes called the **presumed pdf approach**. There are more elaborate methods which solve transport equations to obtain the pdf. The interested reader is referred to Pope (1990) and Dopazo (1994) for details. Although these models are perceived to be more accurate they are currently too computationally expensive to be used in engineering calculations. In contrast, the simple fast chemistry model with presumed β-pdf provides reasonable estimates of temperature and major species in turbulent combusting flows. The main assumption in SCRS is that combustion takes place due to a one-step irreversible reaction with fast chemistry and complete combustion. Fuel and oxygen combine and burn completely on the stoichiometry surface: therefore the model is also known as the **flame-sheet model**. The stoichiometric surface is also the maximum temperature surface in the flame. The main disadvantage of the flame-sheet model is the fact that intermediate products such as CO and H_2 are not predicted. The absence of endothermic dissociation reactions generating the minor species may lead to a substantial overprediction of temperature and major species. A discussion of the modifications to this model to include minor species is available in Peters (1984).

12.21 The chemical equilibrium model

In section 12.8 we discussed how chemical equilibrium is used to calculate species concentrations using simple chemical reactions. The methodology

can be extended to turbulent combustion calculations by considering intermediate reactions. Dedicated **chemical equilibrium calculation programs**, such as CHEMKIN, can be employed to predict equilibrium species concentrations including minor species. Species concentration profiles as a function of mixture fraction can be generated with the aid of such equilibrium programs and used as an alternative to fast chemistry relationships (Peters, 1984; Warnatz *et al.*, 2001).

This method has been successfully used by Kent and Bilger (1973) to predict hydrocarbon flames. Application of the equilibrium model to gas turbine combustors by Jones and Priddin (1978) has shown overprediction of CO and H_2 levels in fuel-rich regions. This is caused by the fact that the local turbulent and diffusion time scales in practical combustor applications are much smaller than the time required to achieve equilibrium. Hence, predictions based on the built-in assumption that the minor species reactions reach equilibrium tend to overestimate minor species levels. As a general rule, the chemical equilibrium model should only be used in situations where the residence time is sufficiently long. However, it is useful since minor species can be predicted and implementation is straightforward. An alternative to the equilibrium model is the partial equilibrium model, which assumes partial equilibrium for some species and non-equilibrium for others. The reader is referred to Eickhoff and Grethe (1979) for further details.

12.22 Eddy break-up model of combustion

Another simple and very efficient model used in combustion calculations is the eddy break-up model due to Spalding (1971). In the eddy break-up model, the rate of consumption of fuel is specified as a function of local flow properties. The mixing-controlled rate of reaction is expressed in terms of the turbulence time scale k/ε, where k is the turbulent kinetic energy and ε is the rate of dissipation of k. The reaction rate is equal to the turbulent dissipation rate, which, for fuel, oxygen and products, may be expressed as follows:

$$\tilde{\bar{\omega}}_{fu} = -C_R\bar{\rho}\tilde{Y}_{fu}\frac{\varepsilon}{k} \tag{12.162}$$

$$\tilde{\bar{\omega}}_{ox} = -C_R\bar{\rho}\frac{\tilde{Y}_{ox}}{s}\frac{\varepsilon}{k} \tag{12.163}$$

$$\tilde{\bar{\omega}}_{pr} = -C'_R\bar{\rho}\frac{\tilde{Y}_{pr}}{(1+s)}\frac{\varepsilon}{k} \tag{12.164}$$

Note that the above expressions describe the Favre-averaged reaction rates.

The eddy break-up model solves one transport equation for the mass fraction of fuel $\tilde{Y}_{fu}$. The individual dissipation rates (12.162)–(12.164) of fuel, oxygen and products are considered, and the model takes the actual reaction rate of fuel to be equal to the slowest of these dissipation rates:

$$\boxed{\tilde{\bar{\omega}}_{fu} = -\bar{\rho}\frac{\varepsilon}{k}\min\left[C_R\tilde{Y}_{fu},\ C_R\frac{\tilde{Y}_{ox}}{s},\ C'_R\frac{\tilde{Y}_{pr}}{1+s}\right]} \tag{12.165}$$

where C_R and C'_R are model constants. Using a different constant for the products allows the existence of the burnt gases to be taken into account. Typical values used in the literature are $C_R = 1.0$ and $C'_R = 0.5$. Since the above expressions contain mass fractions of fuel, oxidant and products, initialisation of these mass fractions is required to start the calculation. In addition to the equation for $\tilde{Y}_{fu}$ a transport equation for mixture fraction $\tilde{\xi}$ is also solved to deduce the product and oxygen mass fractions using relationships such as (12.103), (12.104) and (12.106).

Figures 12.8a and b show the results of Magnussen and Hjertager (1976), who obtained good predictions of the temperature field in furnace configurations with the eddy break-up model. Figure 12.9 shows a further application of the eddy break-up approach combined with the pdf method to account for scalar fluctuations by Gosman *et al.* (1978) in furnace simulations, and again the prediction compares very well with the experimental data.

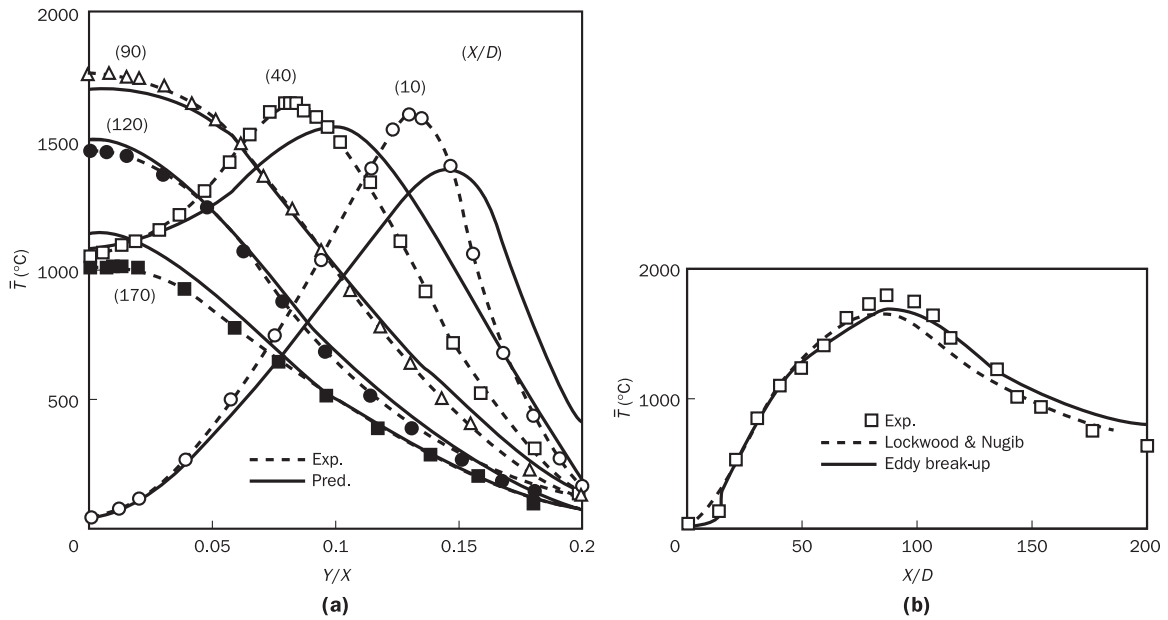



Figure 12.8 Results of the eddy break-up model (Magnussen and Hjertager, 1976): (a) comparison of experimental (Lockwood and Odidi, 1975) and eddy break-up model predictions of local mean temperatures of a city gas diffusion flame ($Re = 24000$); (b) experimental mean temperatures on the axis of the city gas diffusion flame ($Re = 24000$) compared with prediction by Lockwood and Naguib (1975) and the prediction of the eddy break-up model
Source: Magnussen and Hjertager (1976)

The eddy break-up model can also accommodate kinetically controlled reaction terms. When the combustion processes are kinetically controlled, the fuel dissipation rate may be expressed by the Arhennius kinetic rate expression

$$\tilde{\omega}_{fu,kinetic} = -A_1 \bar{\rho}^a \tilde{Y}_{fu}^b \tilde{Y}_{ox}^c \exp(-E_a / R_u \tilde{T}) \tag{12.166}$$

where A_1 is the pre-exponential constant for the Arrhenius reaction rate; a, b and c are model constants; T is the temperature in K; E_a is the activation energy; and R_u is the universal gas constant.

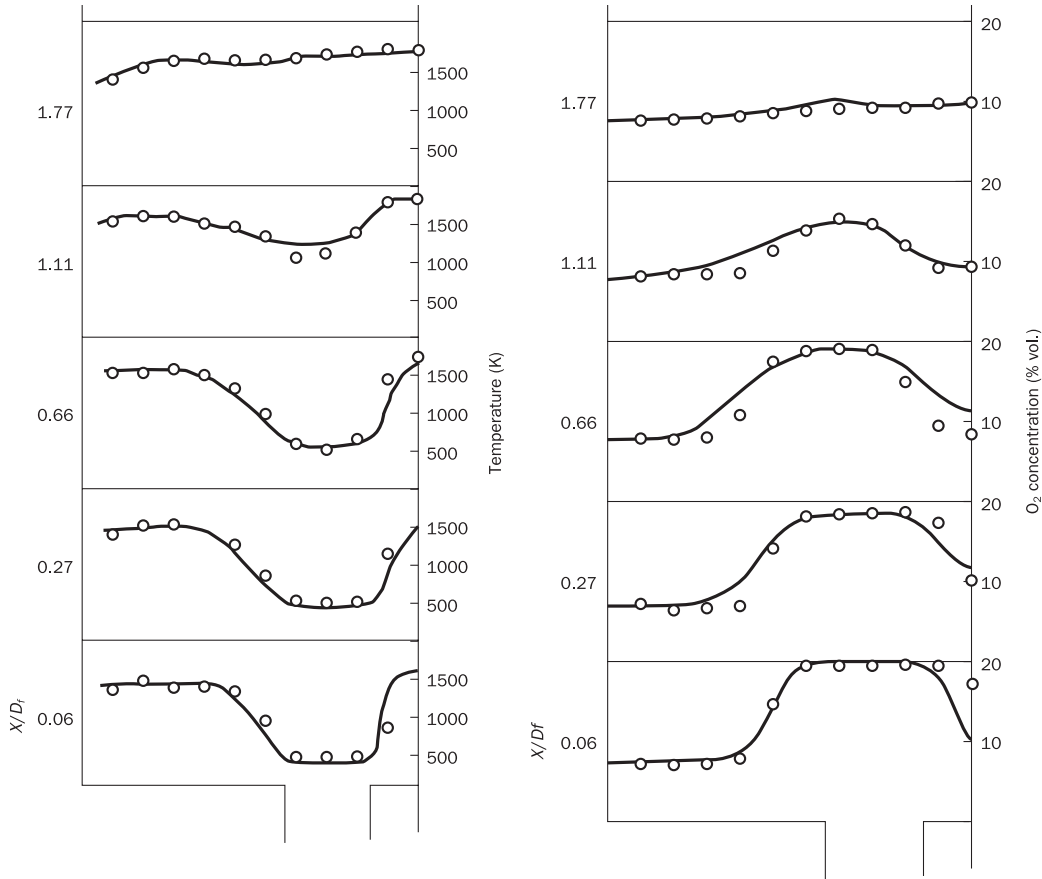



Figure 12.9 Comparison of predictions and experimental data in a furnace simulation (Case 6): radial temperature and oxygen concentration profiles
Source: Gosman *et al*. (1978)

Now the reaction rate of fuel is given by

$$\tilde{\omega}_{fu} = -\min\left[\bar{\rho}\frac{\varepsilon}{k}C_R\tilde{Y}_{fu}, \bar{\rho}\frac{\varepsilon}{k}C_R\frac{\tilde{Y}_{ox}}{s}, \bar{\rho}\frac{\varepsilon}{k}C_R'\frac{\tilde{Y}_{pr}}{1+s}, -\tilde{\omega}_{fu,kinetic}\right] \qquad (12.167)$$

Nickjooy *et al*. (1988) used the above approach by considering a two-step global reaction model which included CO formation to predict combusting flows in axisymmetric combustors. The model constants C_R, C_R', a, b and c used in their study for the different cases considered can be found in the above reference. Figure 12.10 shows the predictions reported by Nikjooy *et al*. (1988).

The eddy break-up model makes reasonably good predictions and is fairly straightforward to implement in CFD procedures. However, due to the dependence of the fuel dissipation rate on the turbulence time scale k/ε, the quality of the predictions depends on the performance of the turbulence model. Where a turbulence model fails to make accurate flow predictions the quality of combustion simulations will, of course, also be limited.

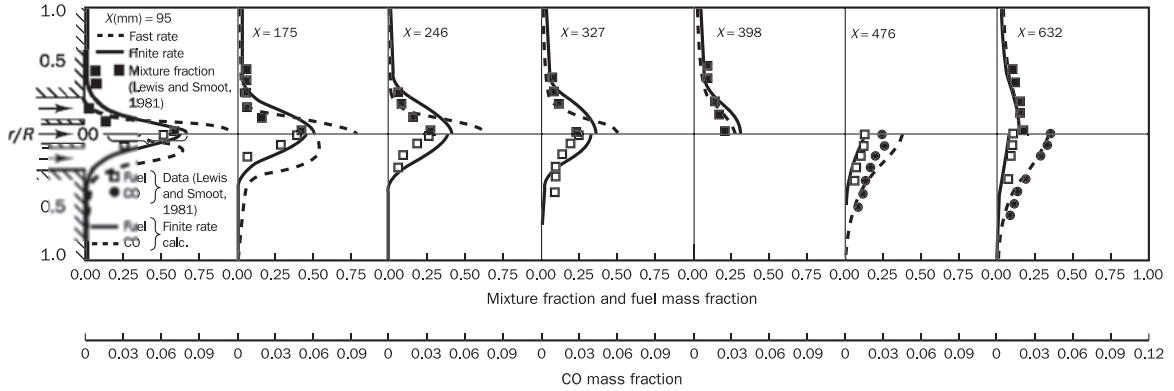



Figure 12.10 A comparison of the calculated and measured mixture fraction (above centre line), fuel and CO (below centre line) for the Lewis and Smoot (1981) experiment
Source: Nikjooy *et al*. (1988)

12.23 Eddy dissipation concept

A modified version of the eddy break-up (EBU) model is known as the eddy dissipation concept (EDC) of Ertesvag and Magnussen (2000). The EDC model attempts to incorporate the significance of fine structures in a turbulent reacting flow in which combustion chemistry is important. A comprehensive review of the EDC model can be found in Magnussen (2005). Without giving the conceptual details the model may be summarised as follows.

In the EDC model the mass fraction occupied by the fine structures is defined as

$$\gamma^* = 4.6\left(\frac{\nu\varepsilon}{k^2}\right)^{1/2} \tag{12.168}$$

Here ν is kinematic viscosity, k and ε are turbulent kinetic energy and dissipation, and 4.6 is a model constant. The reacting fraction of the fine structure is defined as

$$\chi = \frac{\tilde{Y}_{pr}/(1+s)}{\tilde{Y}_{\min} + \tilde{Y}_{pr}/(1+s)} \tag{12.169}$$

where $\tilde{Y}_{pr}$ is the product mass fraction and, as in the EBU model, $\tilde{Y}_{\min} = \min(\tilde{Y}_{fu}, \tilde{Y}_{ox}/s)$. Then the reaction rate for fuel is obtained from

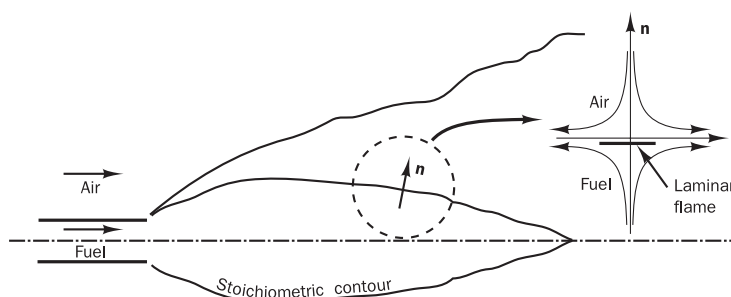
$$\tilde{\dot{\omega}}_{fu} = -\bar{\rho}\frac{\varepsilon}{k}C_{\mathrm{EDC}}\min\left(\tilde{Y}_{fu}, \frac{\tilde{Y}_{ox}}{s}\right)\left(\frac{\chi}{1-\gamma^*\chi}\right). \tag{12.170}$$

Here C_{EDC} is a model constant with a recommended value of 11.2.

12.24 Laminar flamelet model

All the above models except the equilibrium model either use single-step combustion or accommodate only a limited degree of combustion chemistry and do not predict intermediate and minor species. We now discuss the

laminar flamelet model, which is a compromise between simplicity and the need to account for detailed chemistry. The model views the turbulent flame as consisting of an ensemble of stretched laminar flamelets. Figure 12.11 shows a schematic sketch of the laminar flamelet concept. Turbulent flames are described as wrinkled, moving laminar sheets of reaction, and major heat release is considered to occur in narrow regions in the vicinity of stoichiometric surfaces. These are called flamelets, which are considered to be embedded within the turbulent flow. The approach is based on the notion that, if the chemical time scales are much shorter than the characteristic turbulence time scales, the fuel and oxidant react in locally thin one-dimensional structures normal to the stoichiometric contour, as illustrated in the inset of Figure 12.11. These structures are assumed to resemble the flame sheets responsible for combustion in a laminar flame: hence the name **laminar flamelets**. In a turbulent flow the flamelets are considered to be stretched and strained by flow and turbulence. These effects are incorporated in the modelling by including appropriate parameters, as we will see below.

Figure 12.11 Schematic representation of the laminar flamelet concept

The properties of laminar flamelets, e.g. the temperature, density and species mass fractions as a function of the mixture fraction ξ, are evaluated once and for all outside the flow field calculation to yield a so-called **laminar flamelet library**. A library of such relationships used in a calculation is shown later in an example (see Figure 12.16). The library comprises a set of relationships $\phi(\xi)$ between scalar flow properties ϕ and the mixture fraction ξ. Their form is similar to Figure 12.2, but turbulence causes the flames to be stretched, which alters the details of the relationships $\phi(\xi)$. To accommodate the effect of flame stretching due to the flow field in an actual turbulent flame, it is common practice to incorporate as a parameter either the **strain rate** or the **scalar dissipation rate**, both of which are flow properties, into the laminar flamelet library calculations.

A key advantage of the laminar flamelet methodology is that detailed chemistry can be incorporated relatively economically within the calculations, because the library contains information relating to major species, minor species, density and temperature. This enables us to evaluate important aspects of the combustion process such as the formation of pollutants. Without detailed chemistry it is only possible to calculate flow and approximate energy release due to combustion.

In section 12.25 we give an overview of the methods used to generate a laminar flamelet library. In section 12.26 we show how Favre-averaged

values of a scalar variable in a turbulent flow field can be calculated from a knowledge of the values of the mixture fraction ξ and the scalar dissipation rate χ and the pdfs $\tilde{P}(\xi)$ and $\tilde{P}(\chi)$. In sections 12.27 to 12.29 we discuss how the formation of pollutants such as NO can be modelled within the laminar flamelet framework. Finally, in section 12.30 we give an illustrative example to demonstrate the strengths and weaknesses of the laminar flamelet model. We summarise the main aspects of the laminar flamelet method from the work of Peters (1984, 1986), Bray and Peters (1994), Pitsch and Peters (1998), Williams (2000) and Veynante and Vervisch (2002). The interested reader is directed to these publications for further details of laminar flamelet theory for non-premixed combustion.

12.25 Generation of laminar flamelet libraries

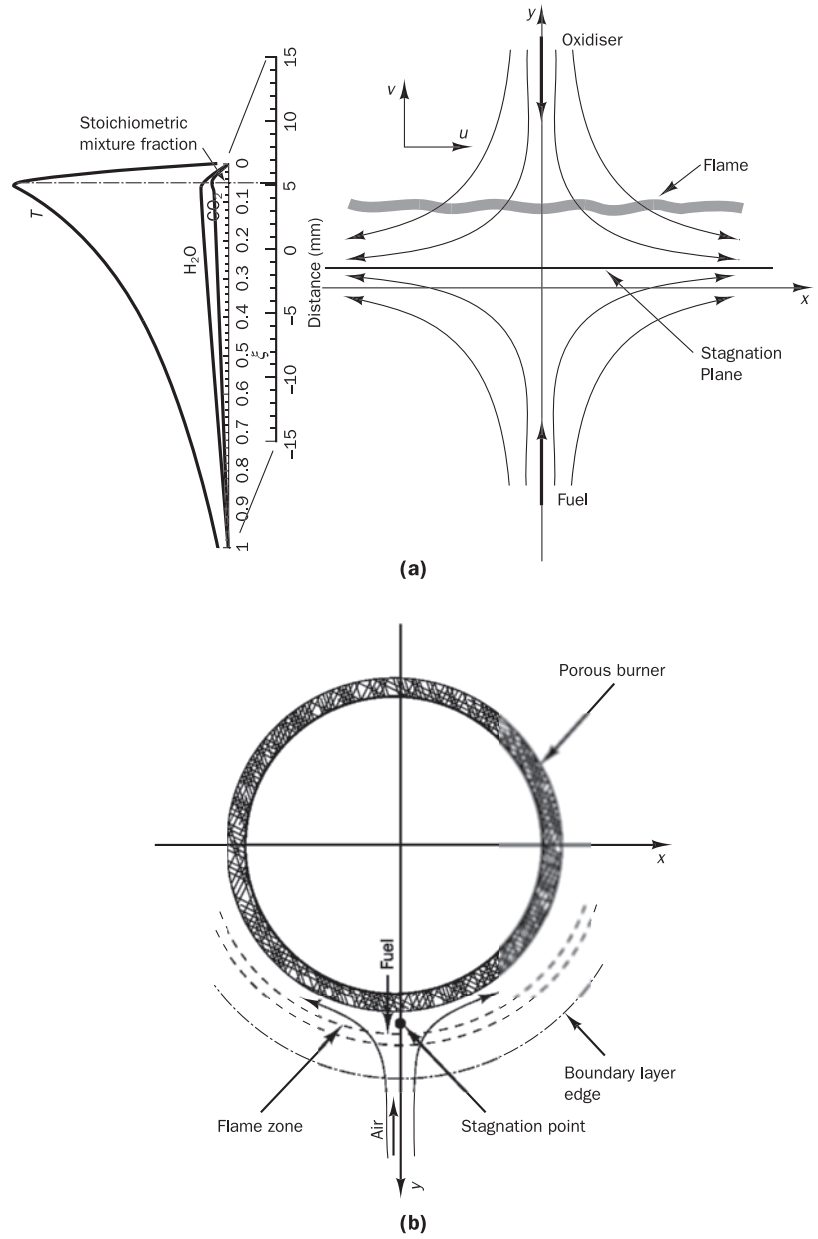
Laminar flamelets are generated by solving the one-dimensional governing equations of non-premixed combustion. The simplest model for counter-flow diffusion flames assumes a steady state, laminar, stagnation-point flow. The governing equations for continuity, momentum, chemical species and energy can be written and solved without much effort. A simple one-dimensional non-premixed combustion situation is shown in Figure 12.12a, where fuel and oxidant streams are opposed jets. There are two main methods of producing laminar flamelet libraries:

- Method 1: the physical (x–y) co-ordinate system is used to solve governing equations for opposed flow diffusion flames
- Method 2: the governing equations of opposed flow diffusion flames are transformed into mixture fraction space and solved

Method 1: Opposed flow diffusion flame configuration

Consider the opposed flow diffusion flame situation shown in Figure 12.12a. Such a flow might be created using two concentric axisymmetric opposing nozzles, one delivering fuel and the other delivering oxidant. The configuration produces an axisymmetric flow field with a stagnation plane between the nozzles as shown. The location of the stagnation plane depends on the velocities of the two streams, and can be adjusted by varying the velocity of one stream and keeping the other constant. A diffusion flame is established between the two nozzles close to the stagnation plane. The position of this thin flat flame depends on the velocities of the two streams and the properties of the fuel and oxidant. Since most fuels require more air than fuel by mass the diffusion flame usually sits on the oxidiser side of the stagnation plane as shown. For calculation purposes the configuration is considered to be infinitely wide and axisymmetric. The co-ordinate system used in the Method 1 formulation is shown in Figure 12.12a. The oxidiser jet is located at $y = \infty$ and the fuel jet at $y = -\infty$. The model assumes the flow to be laminar, stagnation-point flow in cylindrical co-ordinates.

The structure of the flame is obtained by the solution of steady boundary layer equations written along the stagnation streamline (see Smooke *et al.*, 1986; Lutz *et al.*, 1997). The method looks for self-similar solutions that are functions of the co-ordinate perpendicular to the flame, i.e. y only (note

Figure 12.12 (a) An opposed flow flame; (b) Tsuji burner configuration (after Tsuji and Yamaoka, 1967)

that y is the axial co-ordinate in this case). In the usual notation the variables are

$$\rho = \rho(y)$$
$$T = T(y)$$
$$Y_k = Y_k(y),\ k = 1, \ldots, N$$

If y denotes the axial direction and x denotes the radial direction the steady state conservation of mass is as follows:

$$\frac{\partial(\rho u x)}{\partial x} + \frac{\partial(\rho v x)}{\partial y} = 0 \tag{12.171}$$

The radial momentum equation is

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = \frac{\partial}{\partial y}\left[\mu \frac{\partial u}{\partial y}\right] - \frac{\partial p}{\partial x} \tag{12.172}$$

The species conservation equation is

$$\rho u \frac{\partial Y_k}{\partial x} + \rho v \frac{\partial Y_k}{\partial y} + \frac{\partial}{\partial y}(\rho Y_k V_k) - \dot{\omega}_k = 0 \quad k = 1, 2, \ldots, N \tag{12.173}$$

The energy equation is

$$\rho u \frac{\partial T}{\partial x} + \rho v \frac{\partial T}{\partial y} - \frac{1}{c_p}\frac{\partial}{\partial y}\left(k \frac{\partial T}{\partial y}\right) + \frac{\rho}{c_p}\sum_{k=1}^{N} c_{pk} Y_k V_k \frac{\partial T}{\partial y}$$

$$+ \frac{1}{c_p}\sum_{k=1}^{N} h_k \dot{\omega}_k = 0 \quad k = 1, 2, \ldots, N \tag{12.174}$$

The system is closed with the ideal gas law

$$\rho = \frac{p(MW)_{mix}}{R_u T} \tag{12.175}$$

In equations (12.173) and (12.174) the parameter V_k is known as the diffusion velocity, which considers the diffusion of species. For a multi-component system the diffusion velocities are given by (see Lutz *et al.*, 1997; Turns, 2000)

$$V_k = -\frac{1}{X_k(MW)_{mix}}\sum_{j=1}^{N}(MW)_j D_{kj}\frac{\mathrm{d}X_j}{\mathrm{d}x} - \frac{D_k^T}{\rho Y_k}\frac{1}{T}\frac{\partial T}{\partial x} \tag{12.176}$$

where the X_k denote mole fractions, (MW) molecular weights and D_{kj} the multi-component diffusion coefficient, and D_k^T is an equivalent diffusion coefficient of species k into the rest of the mixture. Since calculations of laminar flames attempt to consider detailed chemistry, these fine details are retained in the equations, and detailed transport properties and thermodynamic data calculation routines are also employed in the calculations (see Rogg and Wang, 1997).

In an opposing flow the free stream radial and axial velocities at the edge of the boundary layer are given by $u_\infty = ax$ and $v_\infty = -2ay$, where a is the strain rate. We may now introduce

$$U(y) = \frac{u}{u_\infty} = \frac{u}{ax} \tag{12.177}$$

$$V(y) = \rho v \tag{12.178}$$

Introduction of equations (12.177) and (12.178) transforms the set of equations (12.171)–(12.175) into a system of ordinary differential equations valid along the stagnation-point streamline $x = 0$:

$$\frac{dV}{dy} + 2a\rho U = 0 \tag{12.179}$$

$$\frac{d}{dy}\left[\mu\frac{dU}{dy}\right] - V\frac{dU}{dy} + a(\rho_\infty - \rho U^2) = 0 \tag{12.180}$$

$$-\frac{d}{dy}(\rho Y_k V_k) - V\frac{dY_k}{dy} + \dot{\omega}_k = 0 \quad k = 1, 2, \ldots, N \tag{12.181}$$

$$\frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) - c_p V\frac{\partial T}{\partial y} - \sum_{k=1}^{N}\rho c_{pk} Y_k V_k \frac{dT}{dy} - \sum_{k=1}^{N} h_k\dot{\omega}_k = 0 \tag{12.182}$$

Smooke *et al*. (1986) stated that the appropriate boundary conditions for the fuel (*fu*) and oxidizer (*ox*) streams are as follows:

Fuel stream, $y = -\infty$: $V = V_{fu}$ $\quad U = \sqrt{\rho_{ox}/\rho_{fu}}$ $\quad T = T_{fu}$ $\quad Y_k = Y_{k,fu}$

Oxidiser stream, $y = \infty$: $U = 1$ $\quad T = T_{ox}$ $\quad Y_k = Y_{k,\infty}$

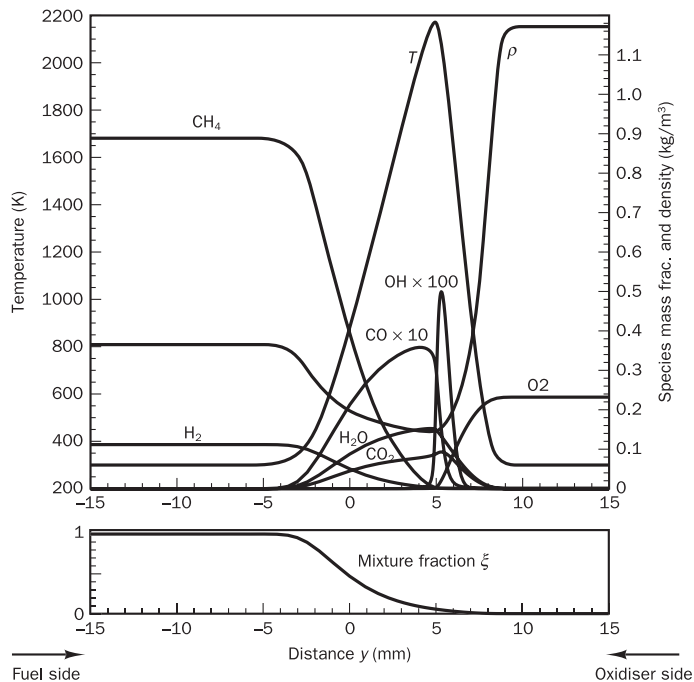
Symmetry plane, $y = 0$: $V = 0$

The distances $y = -\infty$ and $y = +\infty$ refer to fuel and oxidiser steams. In practice the infinite interval is truncated and boundary data are supplied at $y = -L_{fu}$ (the fuel jet) and $y = +L_{ox}$ (the oxidiser jet), where the separation distance is $2L = L_{fu} + L_{ox}$ (see Smooke *et al*., 1986).

Inspection of the above equations shows that the velocity gradient or strain rate a is a parameter of the system of ordinary differential equations (12.179)–(12.182) – the other quantities are the dependent and independent variables and thermodynamic or transport properties. To generate laminar flamelet profiles in the physical space the system is solved for a series of prescribed values of a corresponding to stretch conditions in practical turbulent flows (a = 0.1 to 5000 s^{-1}). Normally, finite difference techniques and Newton's iteration method are used to solve the set of equations.

Another commonly used experimental situation for creating one-dimensional non-premixed flames is the Tsuji burner shown in Figure 12.12b (after Tsuji and Yamaoka, 1967). An alternative formulation of the flamelet equations based on Tsuji's burner configuration, which yields a slightly different set of ordinary differential equations, is also a popular method for laminar flamelet calculations.

There are many programs available for laminar flamelet calculations. RUN1DL (Rogg, 1993; Rogg and Wang, 1997) is widely used for the generation of **laminar flamelet profiles in physical space**. Another computer program for solving counter-flow diffusion flames is the OPPDIF code of Lutz *et al*. (1997). By solving the opposed flow laminar flame problem in physical space the results obtained will give temperature, density, major and minor species as a function of distance from the stagnation point. A typical set of calculation results is given in Figure 12.13, which shows the temperature, density and all species concentrations for an opposed flow laminar flame with a 1:1 mixture of CH_4 and H_2 as fuel for a strain rate of $a_s = 20$ s^{-1}.

Figure 12.13 Laminar flamelet structure of an opposed flow CH_4:H_2 (1:1) laminar flame, air and fuel temperatures 300 K, strain rate $a = 20\ s^{-1}$, fuel left hand side, air right hand side, domain size −15 mm to +15 mm

In combustion procedures we solve the transport equations for mixture fraction and mixture fraction variance. Flame property relationships, such as those in Figure 12.2, lack detailed chemistry. The laminar flamelet model uses the laminar flamelet library as a series of relationships $Y_k = Y_k(\xi)$, $T = T(\xi)$, $\rho = \rho(\xi)$ etc. in the form of a look-up table. This requires us to convert the solutions of (12.179)–(12.182) in physical space $Y_k = Y_k(y)$, $T = T(y)$, $\rho = \rho(y)$ into the mixture fraction co-ordinate. We note that if we go in Figure 12.13 from the fuel stream ($y = -15$ mm) to the oxidiser stream ($y = +15$ mm), the mixture fraction will vary from 1 to 0. In between these extremes the mixture fraction will gradually change. For each flame and each specific value of the strain rate there will be a one-to-one correspondence between the values of the y-co-ordinate and the local mixture fraction $\xi = \xi(y)$. For example, the flame is located in the high-temperature region around $y = +5$ mm, where the mixture fraction will be close to its stoichiometric value, so $\xi_{st} = \xi(y = 5\text{ mm})$. Thus, the results in Figure 12.13 can in principle be replotted using the mixture fraction as the abscissa.

The most common method of evaluating the mixture fraction ξ in combusting flows with an arbitrary number of fuel and oxidiser inlet streams is known as **Bilger's mixture fraction formula** (Bilger, 1988). It is based on the notion that species are consumed or produced during chemical reactions, but *chemical elements are conserved* during reactions. Mass fractions Z of chemical elements (e.g. C, H, O) can be obtained from the mass fractions Y of species containing these elements:

$$Z_j = \sum_{i=1}^{N} \frac{a_{ij} W_j}{MW_i} Y_i \tag{12.183}$$

where MW_i is the molecular weight of species i, W_j is the atomic weight of element j, and a_{ij} is the number of atoms of element j in a molecule of species i. The summation is carried out over all N molecular species. Using these element mass fractions, Bilger's formula for the mixture fraction for the case of a fuel stream that contains only fuel and an oxidiser stream that contains only element O is as follows:

$$\xi = \frac{2Z_C/W_C + Z_H/2W_H - 2(Z_O - Z_{O,ox})/W_O}{2Z_{C,fu}/W_C + Z_{H,fu}/2W_H + 2Z_{O,ox}/W_O} \tag{12.184}$$

Here *fu* and *ox* refer to fuel and oxidant streams.

If both the fuel and oxygen streams contain the elements C, H, O this can be written in a slightly modified form as

$$\xi = \frac{2(Z_C - Z_{C,ox})/W_C + (Z_H - Z_{H,ox})/2W_H - 2(Z_O - Z_{O,ox})/W_O}{2(Z_{C,fu} - Z_{C,ox})/W_C + (Z_{H,fu} - Z_{H,ox})/2W_H - 2(Z_{O,fu} - Z_{O,ox})/W_O} \tag{12.185}$$

Equation (12.185) can be used at every point of the one-dimensional calculation domain to obtain the mixture fraction from profiles of species mass fraction such as those in Figure 12.13. In turn, this enables us to plot temperature, density and species mass fractions as a function of ξ.

These definitions have also been widely used for the interpretation of experimental data in terms of mixture fraction (Masri *et al.*, 1996; Dally *et al.*, 1998b; Barlow *et al.*, 2001) as well as in computational studies (Smooke *et al.*, 1990; Barlow *et al.*, 2000; Hossain and Malalasekera, 2003) and offer a consistent way of defining mixture fractions, especially for obtaining mixture fractions from experimental data. Smooke *et al.* (1986) and Sick *et al.* (1991) have used experimental data for strained laminar flame configurations, e.g. using the Tsuji-type burner or similar to compare with numerical results and validate the methodology for generating laminar flamelet libraries.

Method 2: Flamelet equations in mixture fraction space

The flamelet equations for temperature and species mass fraction of a one-dimensional flame can also be derived by using co-ordinate transformation (of the Crocco type) using mixture fraction ξ as the independent co-ordinate. An opposed-flow diffusion flame is an example of a flame that can be entirely mapped into mixture fraction space. For details of the development of these equations in mixture fraction space the reader is referred to Peters (1984, 1986), who reviewed laminar flamelet concepts for turbulent combustion. Without proof we quote from these references the flamelet equations in mixture fraction space:

$$\text{Species} \quad \rho\frac{\partial Y_k}{\partial t} - \rho\frac{\chi}{2}\frac{\partial^2 Y_k}{\partial \xi^2} - \dot{\omega}_k = 0 \tag{12.186}$$

$$\text{Temperature} \quad \rho\frac{\partial T}{\partial t} - \rho\frac{\chi}{2}\frac{\partial^2 T}{\partial \xi^2} - \frac{1}{c_p}\frac{\partial p}{\partial t} + \sum_{k=1}^{N}\frac{h_k}{c_p}\dot{\omega}_k = 0 \tag{12.187}$$

The boundary conditions are

Fuel stream, $\xi = 1.0$: $T = T_{fu}$ $\quad Y_k = Y_{k,fu}$ $\quad k = 1, \ldots, N$
Oxidiser stream, $\xi = 0.0$: $T = T_{ox}$ $\quad Y_k = Y_{k,ox}$ $\quad k = 1, \ldots, N$

The initial conditions are known temperatures and mass fraction of the species at $t = 0$, i.e. $T = T_{ambient}$, $Y_{O_2} = 0.233$, $Y_{N_2} = 0.767$, Y_{fu} mass fractions (e.g. Y_{CH_4}, Y_{H_2} etc.) are specified using the given fuel composition and mass fractions for all other species are set to zero.

In equations (12.186) and (12.187) χ is the so-called scalar dissipation rate (units s^{-1}) and is a parameter of this set of flamelet equations. The equations are solved in pseudo-transient form, i.e. the transient term is retained, but we are only interested in the final steady state solution; this model is known as the steady laminar flamelet model (SLFM).

Equations (12.186)–(12.187) use a mixture fraction defined on the basis of two streams, fuel and oxidiser, as a co-ordinate whose direction is normal to the stoichiometric surface $\xi = \xi_{st}$, as shown in Figure 12.12a. In turbulent non-premixed combustion, the reaction zone is considered to be in the immediate vicinity of the high-temperature region close to the stoichiometric mixture and convected and diffused with the mixture fraction field (Bray and Peters, 1994).

When the method is used in a CFD computation, influence of the flow field on the flamelet structure is represented in the above equations by the local scalar dissipation rate, defined by

$$\chi = 2D_\xi \left[\left(\frac{\partial \xi}{\partial x} \right)^2 + \left(\frac{\partial \xi}{\partial y} \right)^2 + \left(\frac{\partial \xi}{\partial z} \right)^2 \right] \tag{12.188}$$

where x, y, z are the co-ordinate directions and $D_\xi = \Gamma_\xi / \rho$ is the diffusion coefficient for the mixture fraction. The scalar dissipation is the variable which controls mixing, and represents the gradient of mixture fraction, which is related to strain. When the flame strain increases, the scalar dissipation rate increases. The scalar dissipation rate implicitly incorporates convection–diffusion effects normal to the surface of the stoichiometric mixture (Peters, 1984), and it can be considered as the parameter which describes the departure from equilibrium chemistry. The reciprocal of scalar dissipation χ^{-1} is a measure of diffusive time τ_χ. As this time decreases (i.e. as the value of χ increases) the heat and mass transfer through the stoichiometric surface are enhanced (Veynante and Vervisch, 2002). Furthermore, if χ exceeds a critical value a flame extinguishes as heat losses becomes larger than chemical heat release.

To generate laminar flamelet profiles in mixture fraction space the set of governing partial differential equations (12.186) and (12.187) is solved for given initial and boundary conditions for fuel concentration and temperature for a series of prescribed values of the scalar dissipation rate χ. Different scalar dissipation levels give different flame structures. A very low scalar dissipation rate (i.e. long diffusion time) means that the combustion takes place in conditions that are close to equilibrium, whereas a very high rate means a highly strained flame (close to extinction). Figure 12.16 below shows a set of flamelet relationships produced by this type of calculation. The arrows drawn indicate the direction of increasing scalar dissipation rate.

The resulting flamelet library is a collection of temperature, species and density profiles in the mixture fraction space for different scalar dissipation rates $T(\xi, \chi)$, $Y_i(\xi, \chi)$. The computer program known as FlameMaster by Pitsch (1998) can be used for this.

Finally, it should be noted that equations (12.186)–(12.187) have been derived neglecting many higher-order terms involving convection and cur-

vature along the mixture fraction surface. Nevertheless, they have been used widely (Mauss *et al.*, 1990; Seshadri *et al.*, 1990; Lentini, 1994) to generate flamelets in mixture fraction space. A more accurate formulation which does not rely on the above-mentioned assumptions has been presented by Pitsch and Peters (1998). This new formulation uses a conserved scalar which does not depend on the two-stream formulation for mixture fraction and also allows us to incorporate non-unity Lewis numbers.

An example of laminar flamelet relationships

In this example we consider an experimental dataset available in the literature (and on the Internet, Barlow, 2000) and show how a laminar flamelet library can be generated for particular flame conditions. The resulting state relationships are compared with experimental data to demonstrate the accuracy and usefulness of the laminar flamelet relationships. The calculation of field quantities using the laminar flamelet model will be demonstrated in section 12.30.

We use two popular codes – FlameMaster (Pitsch, 1998) and RUN-1DL (Rogg, 1993; Rogg and Wang, 1997) – to generate flamelets. The FlameMaster code uses Method 2 described above; we have used Method 1 with the RUN-1DL code, i.e. the calculations are performed in physical space. In RUN-1DL flamelets are generated using physical co-ordinates it is necessary to convert the results obtained in physical co-ordinates to mixture fraction space using Bilger's mixture fraction formula (12.184). The chemical mechanism used in the calculation is the detailed mechanism for fuels up to propane by Peters (1993). The same chemical mechanism was used in both codes.

The experimental dataset considered here is the turbulent flame experiment, Flame B, of Barlow *et al.* (2000). This experiment and other datasets have been widely discussed in the 'Turbulent Non-premixed Flames' (TNF) forums. The fuel used is $CO/H_2/N_2$ with a volume ratio of 40/30/30. Flow-dependent effects are represented by two different parameters, strain rate and scalar dissipation rate. The scalar dissipation rate χ is used in mixture fraction space and the strain rate a is used in physical space. The expression which is used to link scalar dissipation rate χ (in axisymmetric cases) to strain rate a is (Peters, 2000)

$$\chi_{st} = \frac{2a_s}{\pi}\exp\{-2[\text{erfc}^{-1}(2\xi_{st})]^2\} \tag{12.189}$$

Here χ_{st} is the scalar dissipation rate at the location where the mixture fraction is stoichiometric and a_s is the corresponding strain rate in physical space.

An alternative expression used for small values of ξ is

$$\chi_{st} = 4a_s\xi_{st}^2[\text{erfc}^{-1}(2\xi_{st})]^2 \tag{12.190}$$

where erfc^{-1} is the inverse of the complementary error function (not the reciprocal). Barlow *et al.* (2000) compared laminar flamelet relationships with their experimental results using two chosen strain rates, $a_s = 10\ \text{s}^{-1}$ and $100\ \text{s}^{-1}$, corresponding to stoichiometric scalar dissipation rates $\chi_{st} = 4.6\ \text{s}^{-1}$ and $46\ \text{s}^{-1}$, respectively. We have repeated their calculations and show the comparison between the laminar flamelet relationships and experimental data at two axial locations of the flame, $x/D = 20$ and 30. The comparison is made in mixture fraction space.

Figures 12.14a and b show the laminar flamelet relationships for temperature and species mole fractions as functions of the mixture fraction. Experimentally measured conditional mean and r.m.s. fluctuation values of temperature and species mole fractions are shown as symbols and error bars, respectively. The solid lines represent the calculation results. Figure 12.14a shows the results obtained using the FlameMaster code and Figure 12.14b shows the results obtained using the RUN-1DL code. It can be seen that the two methods give remarkably similar results. The agreement between flamelet profiles and experimental data is excellent at both locations and well within the experimental error bars. The key inputs for the generation of a flamelet library are the fuel composition and scalar dissipation rates or strain rates. The locations and strain rates used here are those quoted by Barlow *et al.* (2000). No attempt was made to find the most suitable strain rate for a given dataset. The close match between the experimental and calculated values clearly demonstrates the usefulness and accuracy of laminar flamelets as a means of providing state relationships for combustion calculations in a turbulent flame.

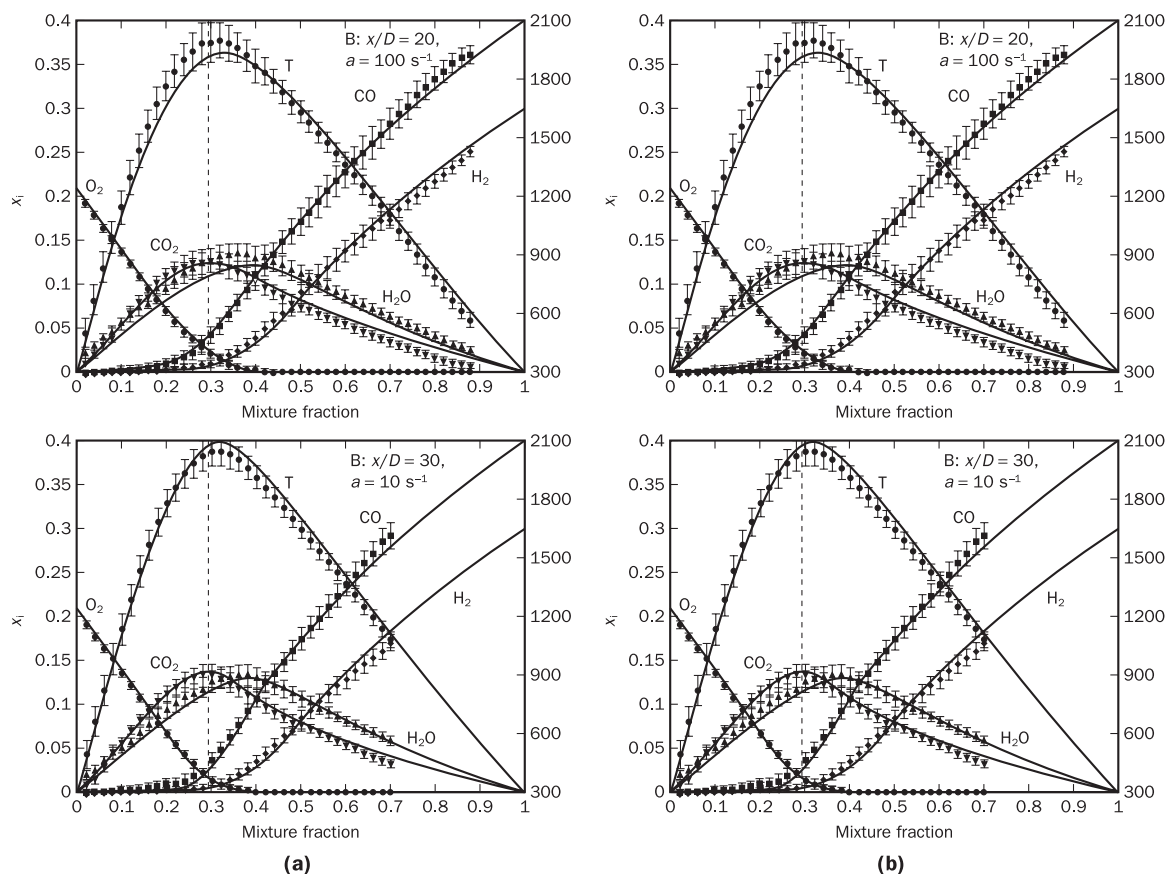



Figure 12.14 Comparison of laminar flamelet relationships with experimental conditional means for temperature, CO_2, H_2O, H_2 and CO for flame B of Barlow *et al.* (2000): (a) FlameMaster calculations; (b) RUN-1DL calculations

12.26 Statistics of the non-equilibrium parameter

In the laminar flamelet model, the thermochemical composition of the turbulent flame is completely determined by two parameters: (i) the mixture fraction ξ and (ii) a non-equilibrium parameter, the scalar dissipation rate χ. In turbulent flow fields, these parameters are statistically distributed. To evaluate mean temperature, density and species mass fractions, it is therefore necessary to know the statistical distribution of the mixture fraction and the scalar dissipation rate in the form of a joint pdf $\tilde{P}(\xi, \chi)$. The mean scalar variables are then evaluated from

$$\tilde{\phi} = \int_0^{\infty}\int_0^1 \phi(\xi, \chi)\tilde{P}(\xi, \chi)\,\mathrm{d}\xi\,\mathrm{d}\chi \tag{12.191}$$

In the current laminar flamelet model formulations, it is assumed that the mixture fraction and the scalar dissipation rate are statistically independent. This assumption simplifies the formulations, because the average value of a scalar variable ϕ in a turbulent flow field is now given by

$$\tilde{\phi} = \int_0^{\infty}\int_0^1 \phi(\xi, \chi)\tilde{P}(\xi)\tilde{P}(\chi)\,\mathrm{d}\xi\,\mathrm{d}\chi \tag{12.192}$$

The pdf for mixture fraction is assumed to be a beta function. The pdf of the scalar dissipation rate is assumed to be a log-normal function, an assumption that is justified by experimental evidence presented in Effelsberg and Peters (1988). Thus, the two independent pdfs are the beta pdf for the mixture fraction

$$\tilde{P}(\xi) = \frac{\xi^{\alpha-1}(1-\xi)^{\beta-1}\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

where parameters α and β are calculated using (12.157), and the log-normal pdf for the scalar dissipation

$$\tilde{P}(\chi) = \frac{1}{\chi\sigma\sqrt{2\pi}}\exp\left(-\frac{1}{2\sigma^2}(\ln\chi-\mu)^2\right)$$

where the parameters μ and σ are related to the first and second moment of χ via

$$\tilde{\chi} = \exp\left(\mu + \frac{1}{2}\sigma^2\right) \tag{12.193}$$

$$\widetilde{\chi''^2} = \tilde{\chi}^2[\exp(\sigma^2) - 1] \tag{12.194}$$

The variance $\widetilde{\chi''^2}$ is not available in this model. Therefore to obtain μ and σ it is necessary to specify $\tilde{\chi}$. The mean scalar dissipation rate $\tilde{\chi}$ is usually modelled as

$$\boxed{\tilde{\chi} = C_\chi \frac{\varepsilon}{k}\widetilde{\xi''^2}} \tag{12.195}$$

Here the model constant C_χ is assigned a value of $C_\chi = 2.0$. The variance σ^2 in the log-normal distribution is set equal to $\sigma^2 = 2.0$. Further details of these two model constants can be found in Peters (1984). By calculating $\tilde{\chi}$ from equation (12.195) and with the specified value of σ^2, the pdf $\tilde{P}(\chi)$ can be evaluated.

A different expression proposed by Bray and Peters (1994) for the scalar dissipation rate is

$$\boxed{\tilde{\chi} = C_\chi \frac{\varepsilon}{k}(\Delta\xi_{st})} \tag{12.196}$$

With $C_\chi = 2.0$ and $(\Delta\xi_{st})$ as the range of mixture fraction values within which the scalar dissipation process is concentrated, then $(\Delta\xi_{st})$ is an indication of the thickness of the reaction zone. With the reaction zone centred at the location $\xi = \xi_{st}$, the thickness of the reaction zone is approximated as

$$\boxed{(\Delta\xi_{st}) \approx 2\xi_{st}} \tag{12.197}$$

where ξ_{st} is the stoichiometric mixture fraction. This expression has proved to be successful in predicting lift-off heights in jet flames.

To calculate the mean of a variable it is required to evaluate the integral (12.192). Integration of the β-pdf was discussed before in section 12.20. The same procedure applies here. Integration with the log-normal pdf $\tilde{P}(\chi)$ can be carried out using the approximation of Lentini (1994). More details of the origin and the theory of laminar flamelet models can be found in Peters (1984, 1986), Bray and Peters (1994) and Veynante and Vervisch (2002).

12.27 Pollutant formation in combustion

NO_x, SO_x, heavy-metal compounds and particles are the main pollutants formed in combustion. The amount and nature of pollutants formed depend on the type of fuel, combustion equipment and temperatures used in the combustion process.

Among the combustion gases:

- NO_x and SO_x contribute to acid rain and other environmental effects
- CO_2 contributes to the greenhouse effect

NO_x is of particular concern:

- It has an impact on the formation of ozone
- It is linked to the formation of smog
- It is directly responsible for acid rain
- It has a considerable impact on human health since it causes conditions such as acute respiratory diseases and respiratory infections

The species NO, NO_2 and N_2O are collectively known as NO_x and result from two sources:

- Molecular N_2 (air)
- Fuel-bound nitrogen

Formation mechanisms of NO_x in the combustion process are further classified as follows:

- *Thermal NO_x*: This is formed by the direct reaction of molecular nitrogen and oxygen at high flame temperatures, the formation of which is described by the Zel'dovich mechanism. This will be discussed further below.
- *Prompt NO_x*: This is formed by oxidation of atmospheric nitrogen in the flame front. Prompt NO_x formation occurs by the reaction of molecular nitrogen with hydrocarbon free radicals in oxygen-deficient regions of the flame. The mechanism often used to describe prompt NO formation is called the Fenimore (1970) mechanism, which considers the rapid formation of NO long before the time required to form thermal NO. More detailed description of the mechanism and further details can be found in Turns (2000) and Kuo (2005).
- *Fuel NO_x*: This arises due to the presence of nitrogen in the fuels such as heavy oil and coal. Fuel NO_x formation reactions are not well understood. The principal reaction process appears to be devolatilisation of fuel nitrogen, forming intermediate compounds such as NO, NO_2, NH_3 and HCN. Subsequent N_2 or NO formation depends strongly on the fuel/air ratio. In general, a fuel with lower nitrogen content typically produces less total NO_x than a fuel with higher nitrogen content. For further details see Kuo (2005).

In addition to these, there is a further route known as the N_2O intermediate mechanism, which is only important in high pressures. The relative contribution of different pathways depends on fuel type, temperature, pressure and residence time. Out of these, thermal NO appears to be the main contributor in most practical combustion situations. For a more detailed discussion of NO_x formation in combustion the reader is referred to Turns (1995, 2000), Warnatz *et al*. (2001) and Kuo (2005).

12.28 Modelling of thermal NO formation in combustion

Among the NO_x components NO is the main pollutant. Thermal NO is formed by N_2 in the air stream reacting with O_2 at high temperature. The reaction mechanism of thermal NO formation is well known as the Zel'dovich mechanism. The key reactions are

$$(1) \quad O + N_2 \overset{k_1}{\longleftrightarrow} NO + N \tag{12.198}$$

$$(2) \quad N + O_2 \overset{k_2}{\longleftrightarrow} NO + O \tag{12.199}$$

$$(3) \quad N + OH \overset{k_3}{\longleftrightarrow} NO + H \tag{12.200}$$

Forward and backward reaction rate constants for these three key reactions are given in Table 12.4 (Turns, 2000).

Table 12.4

Reaction	*Forward reaction rate constant (m^3/(kmol)(s))*	*Backward reaction rate constant (m^3/(kmol)(s))*
1	$k_{1,f} = 1.8 \times 10^{11} \exp(-38370/T)$	$k_{1,b} = 3.8 \times 10^{10} \exp(-425/T)$
2	$k_{2,f} = 1.8 \times 10^{7} T \exp(-4680/T)$	$k_{2,b} = 3.8 \times 10^{6} T \exp(-20820/T)$
3	$k_{1,f} = 7.1 \times 10^{10} \exp(-450/T)$	$k_{1,b} = 1.7 \times 10^{11} \exp(-24560/T)$

Compare the reaction rate constant of reaction (1) with the reaction rate constant of a typical intermediate combustion reaction $H + O_2 \longleftrightarrow OH + O$ (see section 12.10). This reaction has a forward reaction rate constant of $k_f = 2.00 \times 10^{14} \exp(-70.3/T) \mathrm{cm}^3/\mathrm{mol.s}$ (see Kuo, 2005), requiring an activation energy of −585 kJ/kmol. Reaction (1) of the above mechanism by contrast has a very high activation energy ($E_a = -319027$ kJ/mol) and is sufficiently fast only at high temperatures. It has the slowest rate of the reaction in the Zel'dovich mechanism and therefore it is the rate-limiting reaction in the formation of NO. Because of this high-temperature dependence it is a common practice to assume that the thermal NO mechanism is unimportant below 1800 K. Thermal NO reactions (1)–(3) are much slower than other combustion reactions, so NO formation takes too long to achieve equilibrium. Hence, NO concentrations cannot be calculated using equilibrium chemistry. The rate of conversion is governed by the gas temperature (exponentially), residence time of the gas in high-temperature regions, and excess air levels (affecting oxygen availability). The use of laminar flamelet concepts for combustion assumes relatively fast chemistry. Even in the absence of other NO production mechanisms the inclusion of the above NO equations in a laminar flamelet library calculation does not produce experimentally observed NO levels (see Drake and Blint, 1989; Sanders and Gökalap, 1995; Vranos *et al.*, 1992). Laminar flamelet modelling of thermal NO requires some special treatment, which is explained below. As noted above, the other two principal mechanisms of NO formation are prompt (or Fenimore) NO formation and fuel-bound NO formation. These are not included in the method outlined below.

12.29 Flamelet-based NO modelling

As explained earlier, attempts to obtain concentrations of thermal NO from a flamelet library can be inaccurate. One way around this problem is to include an accurate account of the effects of residence time on NO formation by using the Zel'dovich mechanism in conjunction with a transport equation for the mass fraction of NO:

$$\frac{\partial}{\partial t}(\bar{\rho}\tilde{Y}_{NO}) + \frac{\partial}{\partial x_j}(\bar{\rho}\tilde{u}_j\tilde{Y}_{NO}) = \frac{\partial}{\partial x_j}\left(\frac{\mu_{eff}}{\sigma_{NO}}\frac{\partial \tilde{Y}_{NO}}{\partial x_j}\right) + \bar{\dot{\omega}}_{NO} \qquad (12.201)$$

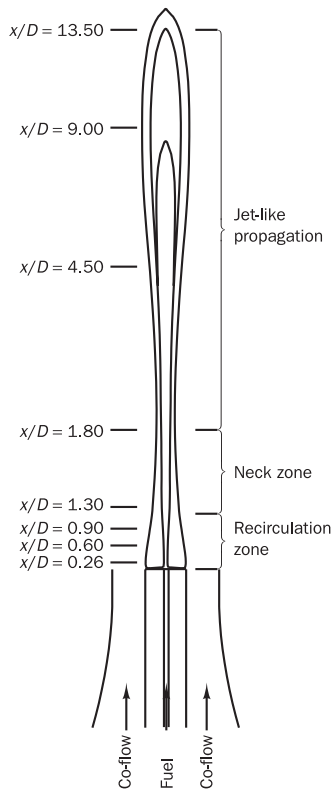
where σ_{NO} is the turbulent Schmidt number, usually assigned a value of 0.7. The average source term $\bar{\dot{\omega}}_{NO}$ of the transport equation represents the mean rate of production of NO, which can be evaluated using the pdf approach:

$$\bar{\dot{\omega}}_{NO} = \bar{\rho}\int_0^{\infty}\int_0^{1} \dot{\omega}_{NO}(\xi, \chi)\tilde{P}(\xi)\tilde{P}(\chi)\,\mathrm{d}\xi\,\mathrm{d}\chi \qquad (12.202)$$

where $\tilde{P}(\xi)$ and $\tilde{P}(\chi)$ are pdfs introduced in section 12.26. The NO source term $\dot{\omega}_{NO}(\xi, \chi)$ can be generated using the rate constants of equations (12.198)–(12.200) and (12.69) and (12.71). It is stored in the laminar flamelet library alongside the flamelet relationships for temperature $T(\xi, \chi)$ and species mass fraction $Y_i(\xi, \chi)$.

12.30 An example to illustrate laminar flamelet modelling and NO modelling of a turbulent flame

We consider an experimental flame configuration for which a set of well-documented experimental data is available and show the predictions of temperature, major and minor species and NO using the laminar flamelet calculation methodology explained in the above sections. The problem considered here is the bluff-body CH_4/H_2 flame experimentally studied by Dally *et al*. (1996, 1998b). This experimental flame is designated as HM1 in the dataset. The experimental data were provided by Masri (1996) via the Sydney University website. The bluff body considered here has an outer diameter of $D_B = 50$ mm and a concentric fuel jet diameter of $D_j = 3.6$ mm. The fuel composition is 50% by volume CH_4 and 50% by volume H_2. The mean velocity of the fuel jet is 118 m/s and the mean air velocity is 40 m/s. A schematic diagram of the flame and important zones is shown in Figure 12.15. Further experimental details are available in the aforementioned references.

Figure 12.15 Schematic drawing of the bluff body stabilised flame and measurement locations

Computational details

The governing differential equations (continuity, momentum, turbulence variables k and ε, mixture fraction, mixture fraction variance) were solved in Favre-averaged form by means of an in-house finite volume code based on the SIMPLE algorithm. The k–ε turbulence model was used for turbulence closure. In section 3.7.2 it was noted that the standard k–ε model overpredicts the decay rate and the spreading rate of a round jet. Several modifications to

the k–ε model are available for correcting this problem (McGuirk and Rodi, 1979; Pope, 1978). In this study the value of constant $C_{1\varepsilon}$ in the ε-transport equation is modified from 1.44 to 1.60 following the work of Dally *et al.* (1998b) and Hossain *et al.* (2001). Following the International Workshop on Measurements and Computations of Turbulent Nonpremixed Flames (TNF) this modification is now recommended practice for modelling bluff-body flames.

The computational domain has a size of 170 mm in the radial direction and 216 mm in the axial direction. A grid arrangement with 89 radial and 99 axial nodes was used in all the calculations presented here. Sensitivity tests were carried out to establish that the grid was sufficiently fine to give accurate results. A fully developed velocity profile was specified as the boundary condition at the air and fuel inlets.

Prediction of flame structure

The laminar flamelet library for different scalar dissipation rates was generated using the RUN-1DL program mentioned earlier. The calculations presented here have been performed using unity Lewis number. Figure 12.16 shows examples of the laminar flamelet relationships for temperature, CO_2, H_2O and OH mass fractions. Turbulent combustion calculations are carried out

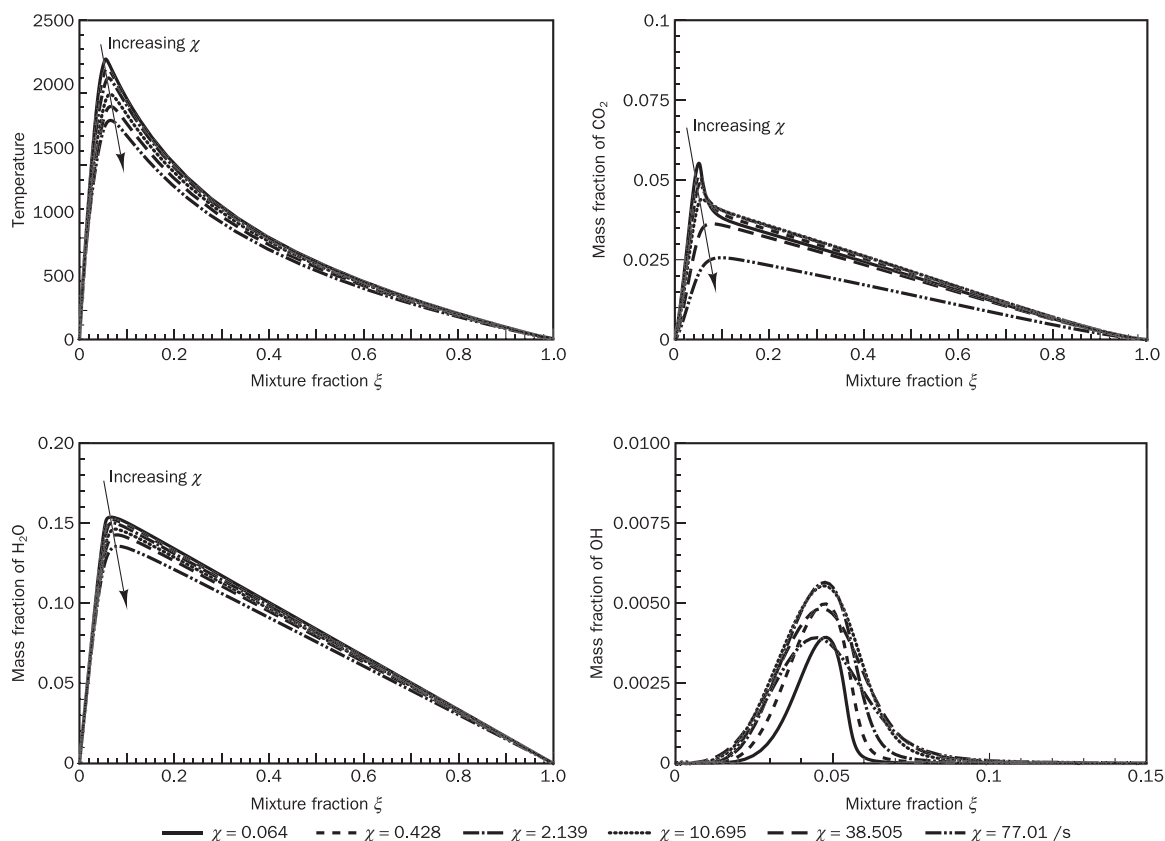



Figure 12.16 A sample of laminar flamelet profiles

by following the methodology explained in sections 12.25 and 12.26. Local values of the scalar dissipation rate were obtained using the approximate relationship (12.196)–(12.197). The library stores relationships at discrete values of scalar dissipation rate χ. Local values of χ are likely to be intermediate between those of the library. The appropriate flamelet relationships required in the calculation process are obtained by linear interpolation of values from neighbouring library look-up tables.

To illustrate the effectiveness of the laminar flamelet model we present in Figures 12.17–12.26 a comparison of numerical predictions with a selection of experimental data in the near field of the burner (i.e. in the region closest

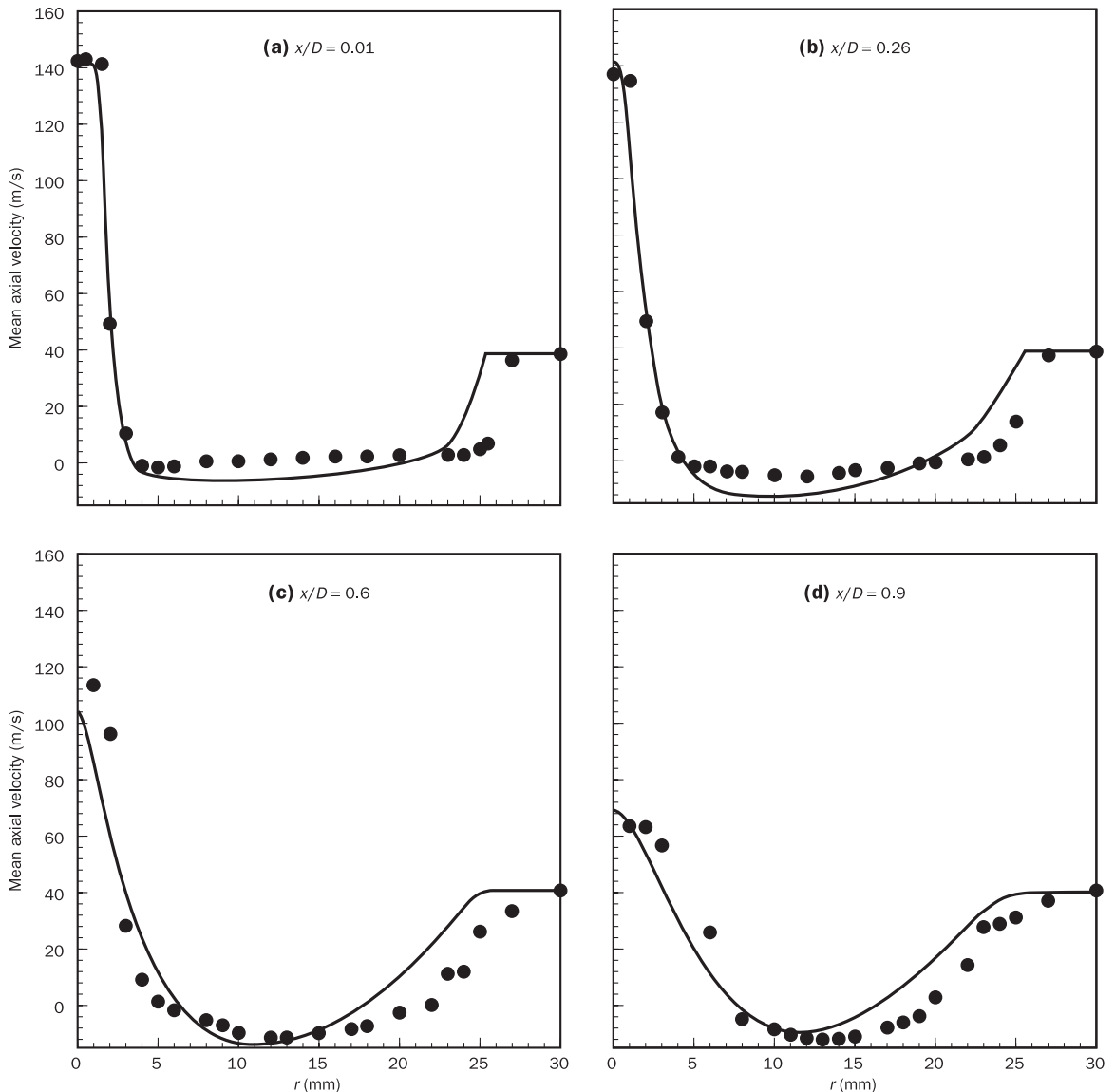



Figure 12.17 Mean axial velocity profiles: ●, measurements; ———, calculations

to the inlets). To start, we note that it is essential that the flow field is adequately represented in the simulation, because the mixture fraction statistics completely determine the thermochemical state of the flame. It is even more important for this CH_4/H_2 flame, because it has a low stoichiometric mixture fraction ($\xi_{st} = 0.05$). As a result, a small error in the flow field will lead to a large error in temperature, species concentrations and location of the flame front. Figures 12.17 and 12.18 show a comparison between measured and computed radial profiles of mean axial and radial velocities plotted at different near-field locations. The mean axial velocity measurements are

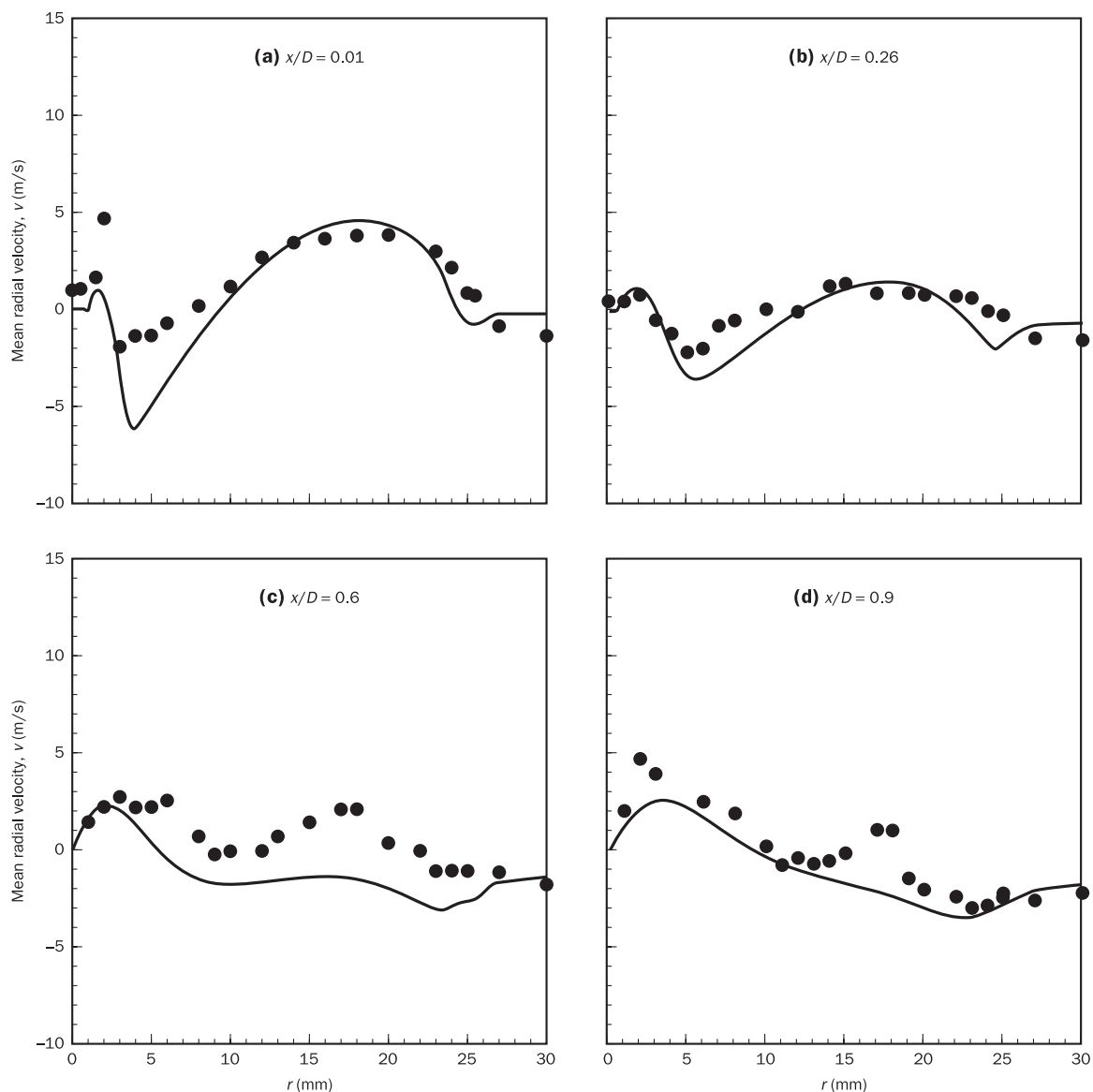



Figure 12.18 Mean radial velocity profiles: ●, measurements; ———, calculations

well predicted. The mean radial velocity measurements, which are much smaller in magnitude, are also well reproduced at all near-field locations. There is some deviation at locations further downstream (not shown here). The mean radial velocity measurements show some scatter, which may be attributed to the inaccuracies in the radial component measurements, as discussed in Dally *et al*. (1998b).

Figure 12.19 shows the radial mixture fraction profiles at four axial locations. The mixture fraction profiles are in excellent agreement with measurements in the near field ($x/D < 0.9$). Here D is the bluff-body diameter. Further

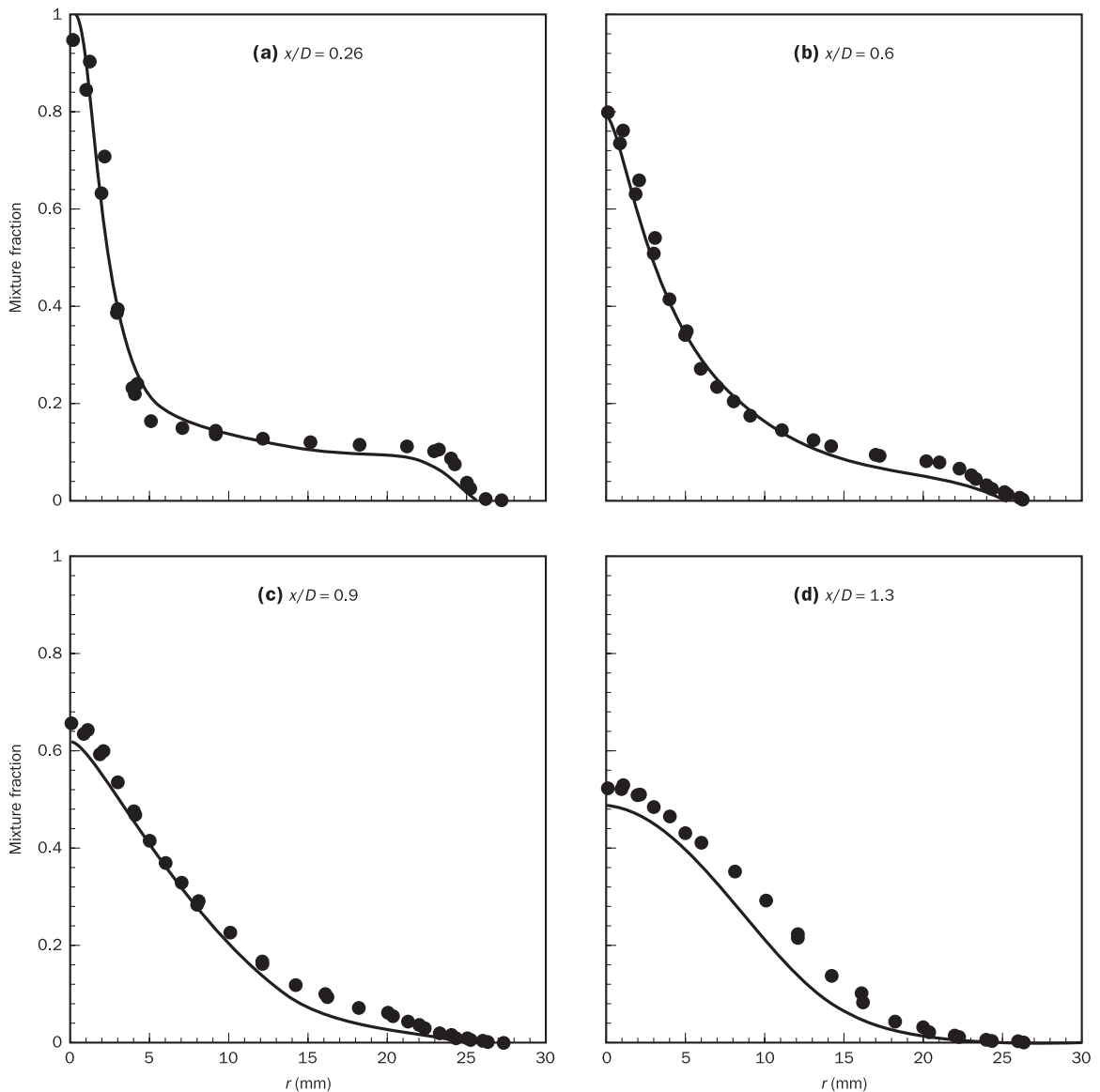



Figure 12.19 Mean mixture fraction profiles: ●, measurements; ———, calculations

downstream ($x/D > 1.3$), the mixture fraction near the axis is underpredicted (not all the results are shown here). This error can be attributed to the known overestimation of the spreading rate of the round jet by the k–ε turbulence model as explained earlier. Clearly, the decay rate is not well represented even after modifying the constant $C_{1\varepsilon}$. Radial profiles of the mixture fraction variance are shown in Figure 12.20. Mixture fraction variance is slightly overpredicted in the near field ($x/D < 1.3$). However, the location of the peak in the near field is well reproduced.

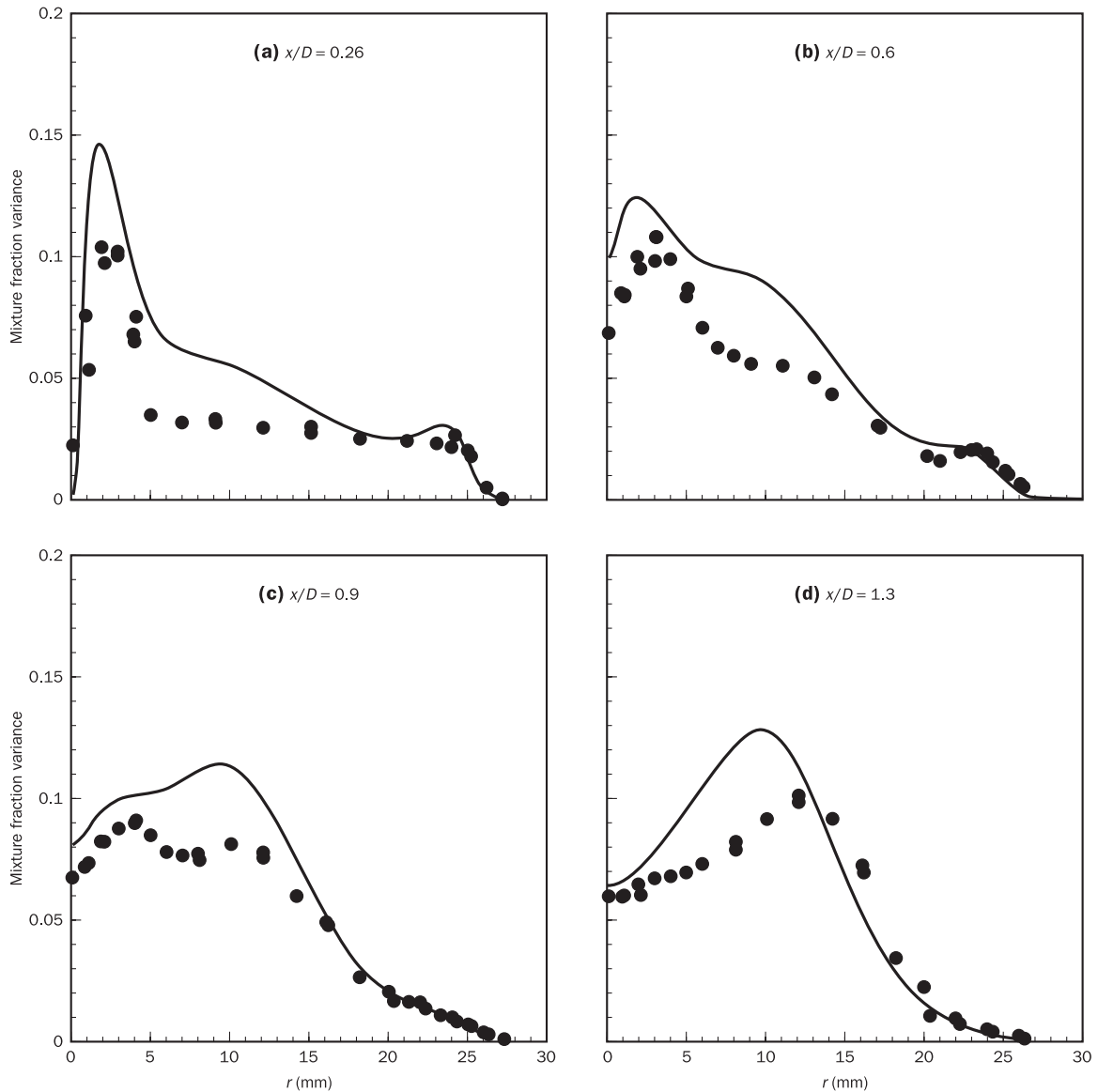



Figure 12.20 Mean mixture fraction variance profiles: ●, measurements; ———, calculations

Radial temperature profiles are shown in Figure 12.21. The mean temperature distribution is fairly well predicted by the laminar flamelet model. Further downstream results show a slight overprediction. It is worth mentioning that a significant overprediction at $x/D = 0.26$ was also observed by Dally *et al*. (1998a). They noted that this might not be due to shortcomings of the simulation, but caused by averaging effects in the temperature measurements as a result of the intermittency in the flame at these locations.

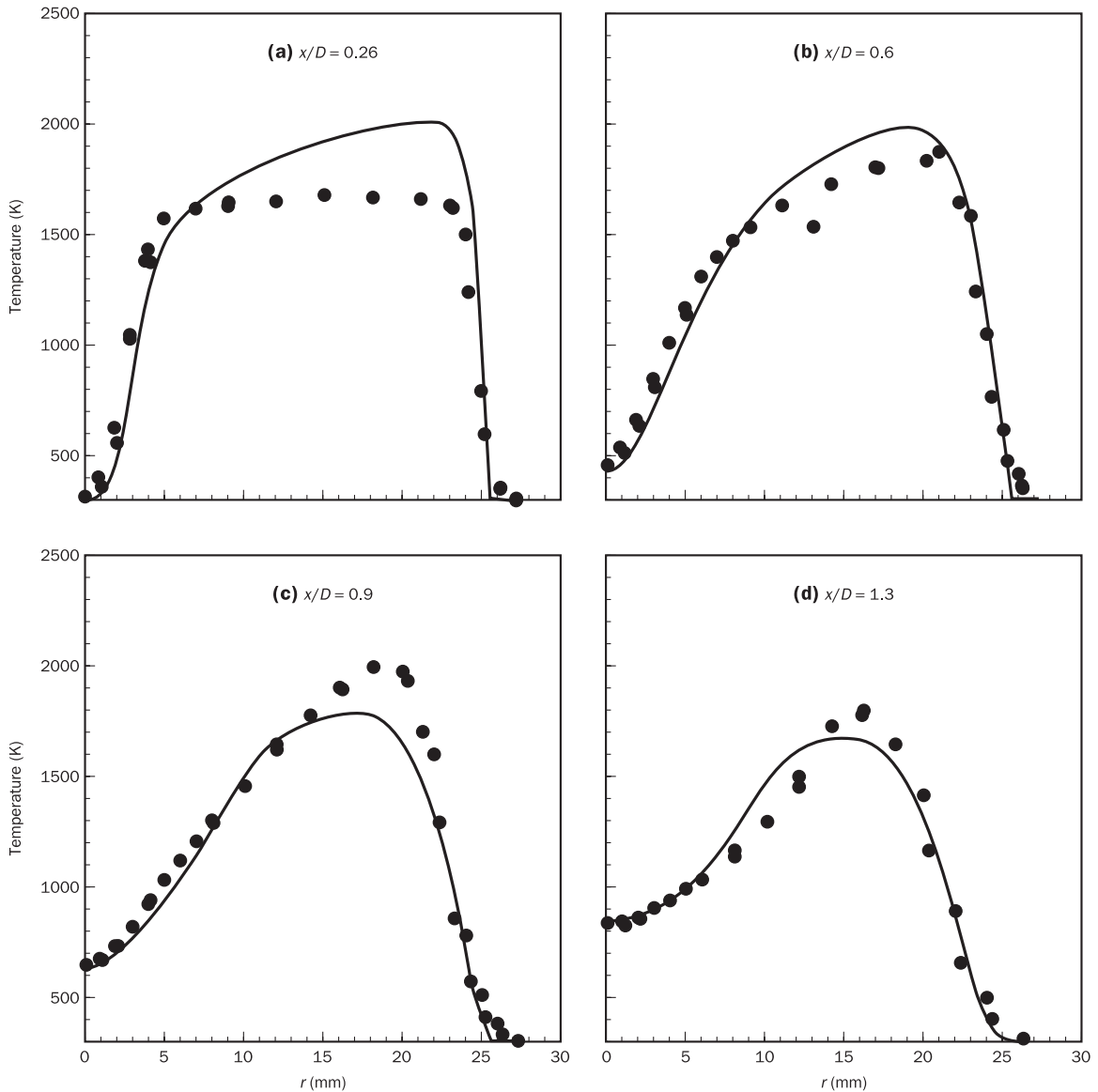



Figure 12.21 Mean temperature profiles: ●, measurements; ———, calculations

Radial distributions of the mass fractions of major species CO_2 and H_2O are shown in Figures 12.22 and 12.23 respectively. Again reasonable agreement with the experimental data is obtained. As mentioned earlier, the inclusion of detailed chemistry in the laminar flamelet calculation allows us to predict minor species such as OH and CO. Radial profiles of the mass fraction of OH, which is formed by dissociation in the high-temperature region of the flame, are shown in Figure 12.24. The experimental data show that the reaction zone of OH is thin in the near field. The thickness of the reaction zone increases further downstream. This illustrates that the flamelet model

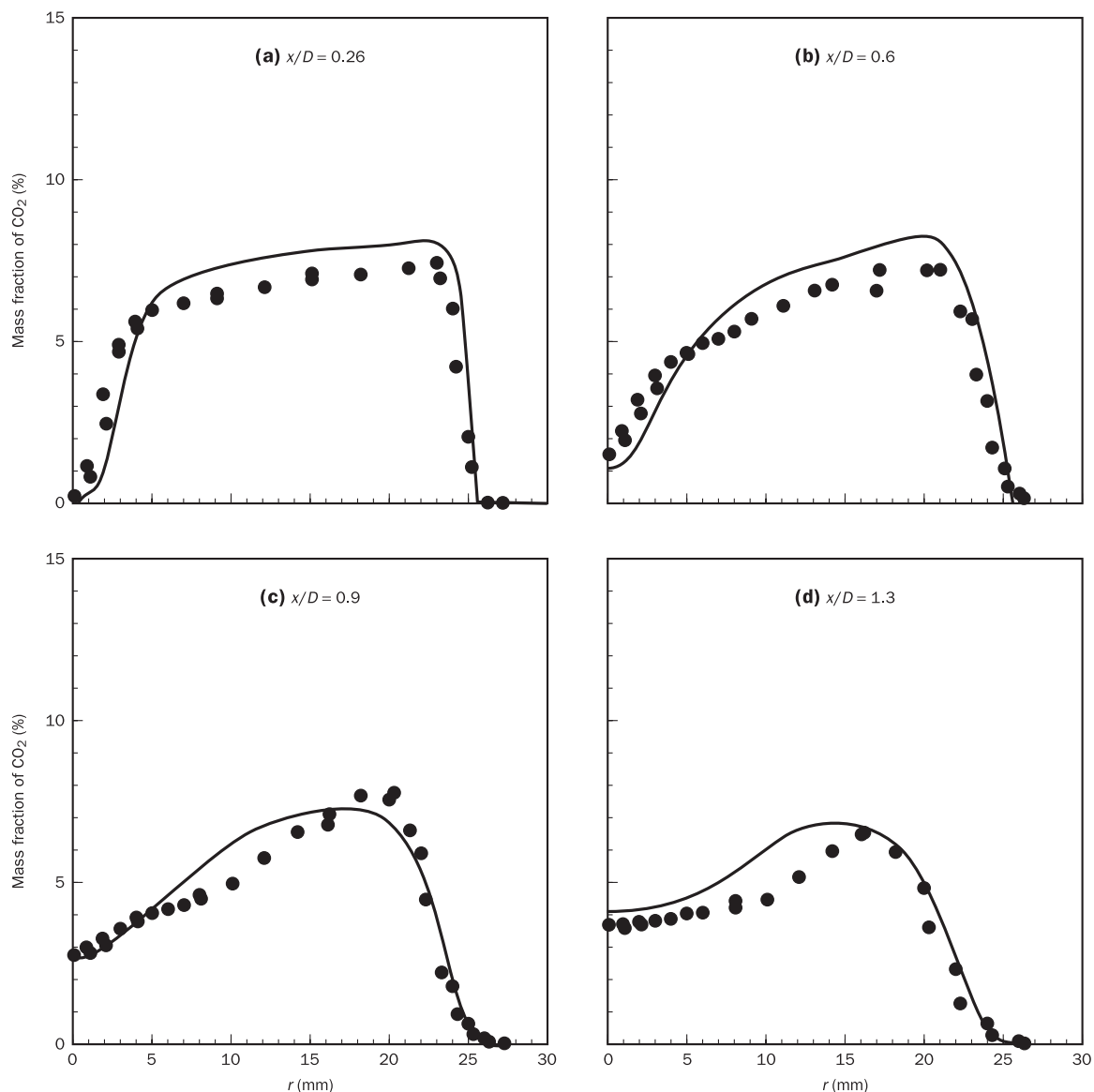



Figure 12.22 Mean CO_2 mass fraction profiles: ●, measurements; ———, calculations

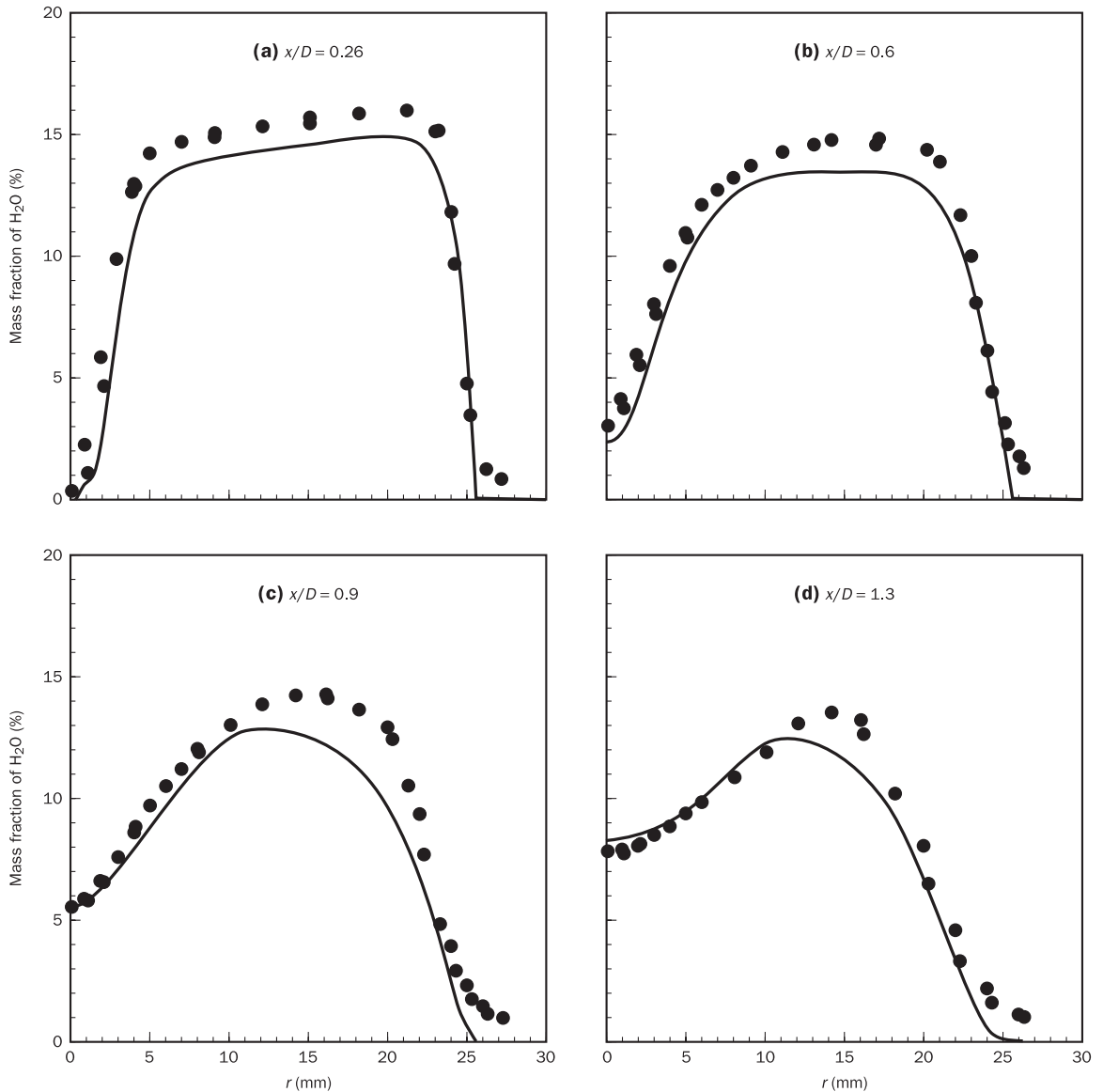



Figure 12.23 Mean H_2O mass fraction profiles: ●, measurements; ———, calculations

is able to capture the general trends of increasing thickness of the OH reaction zone and the rate of decay of OH in the downstream direction.

Prediction of NO

We have also applied the thermal NO prediction procedure explained in section 12.29. Figure 12.25 shows the NO source terms $\bar{\omega}_{NO}/\bar{\rho}$ as a function of mixture fraction for different stretch conditions. The figure shows that the source term is very sensitive to the scalar dissipation rate. At $\chi = 0.064\ s^{-1}$,

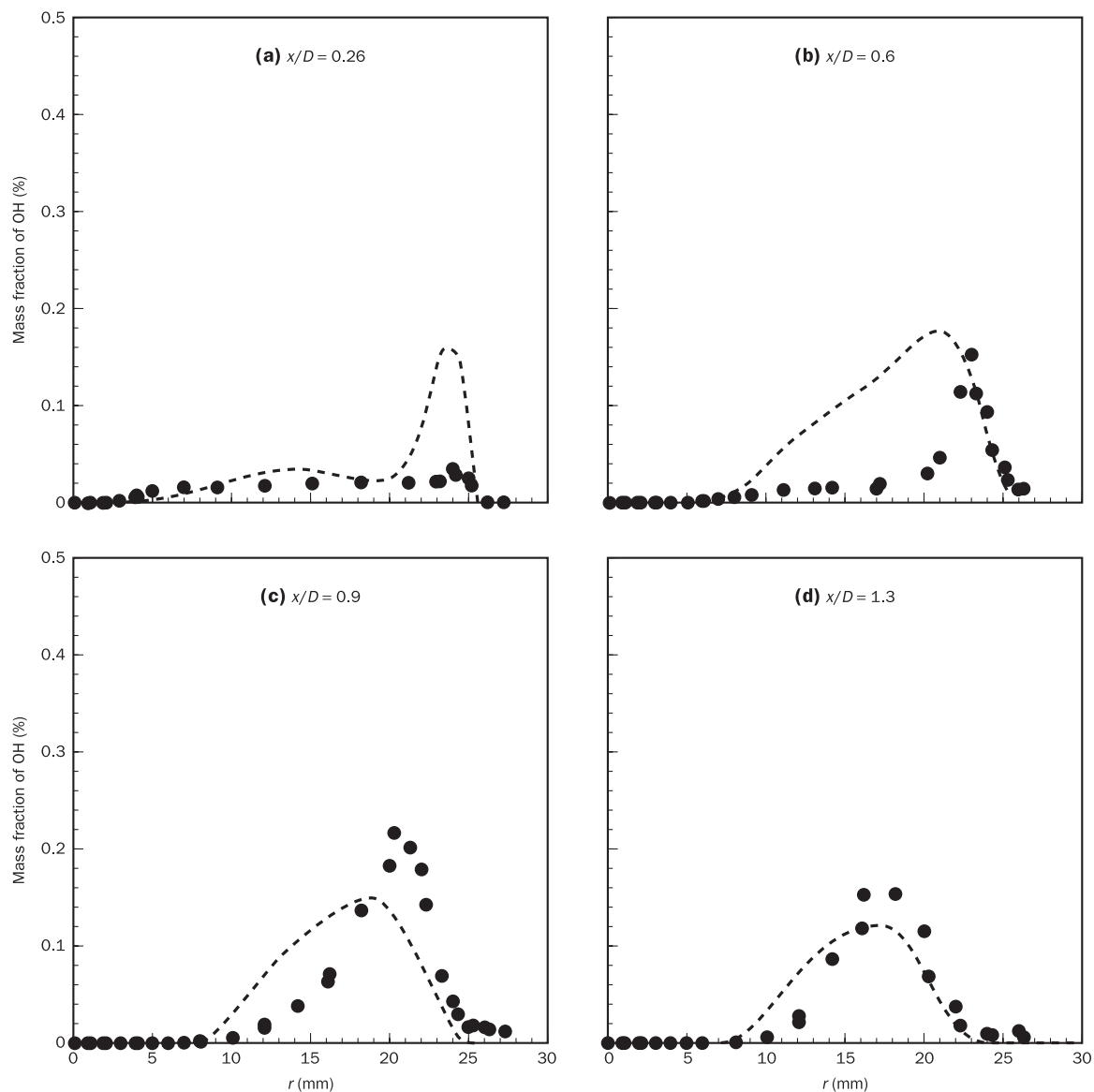
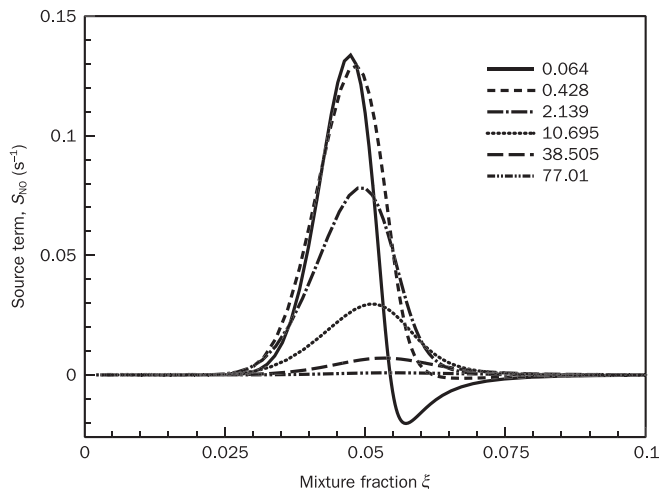



Figure 12.24 Mean OH mass fraction profiles: ●, measurements; ------, calculations

the source term is negative in the fuel-rich zone ($\xi > 0.055$), indicating consumption of NO. At $\chi = 0.428$ s^{-1}, the negative zone of the source term almost vanishes, but the peak value remains constant. At scalar dissipation rates higher than $\chi = 0.428$ s^{-1}, the decrease of temperature is more prominent and this reduces the source term rapidly. As the scalar dissipation rate increases further to $\chi = 77.01$ s^{-1} the formation of NO has almost completely ceased.

Figure 12.25 Flamelet profiles of the NO source term for different scalar dissipation rates

The comparison between the predictions and the measurements for the radial mass fraction of NO is shown in Figure 12.26. The flamelet model seems to underpredict the NO profiles, but it should be noted that we have only considered thermal NO formed through the Zel'dovich mechanism. Even though Chen and Chang (1996) have shown that the Zel'dovich mechanism is the dominant pathway for the production of NO in a turbulent CH_4/H_2 jet flame, it appears that calculations conducted with the thermal NO mechanism alone do not give correct NO levels in this flame. This is further illustrated in this example by considering a more elaborate NO_x mechanism. Shown by dashed lines in Figure 12.26 is a repeat of the NO_x calculation using the GRI 2.11 mechanism, which consists of 49 species and 279 reactions. This mechanism includes reactions to represent both thermal and prompt NO_x chemistry. It can be seen that the dashed lines are in better agreement with the experimental data than the thermal-only predictions (solid lines). The trend is also well predicted. There is still some degree of over-prediction further away from the burner exit. This could be due to discrepancies in temperature predictions seen earlier. Given that the NO mass fractions are very small, the degree of agreement from the GRI 2.11 predictions seen here could considered to be very good (see Murthy *et al.*, 2006, for further details).

In summary, we have illustrated how the laminar flame model can be used to predict important flame properties such as temperature, major species, minor species and NO. The predictions were compared against reported experimental data. In the laminar flamelet model, the mean temperature, density and composition in the turbulent field are obtained by appropriate averaging of the flamelets. The concentration of NO, however, is calculated by solving its own transport equation with a source term obtained from the flamelet library. The flow field is reasonably well reproduced in the calculation. The temperature and concentration of major and minor species are also well reproduced by the flamelet model with unity Lewis number. The prediction of NO is not as good as other species, and it is suggested that inclusion of other NO production mechanisms can improve NO predictions.

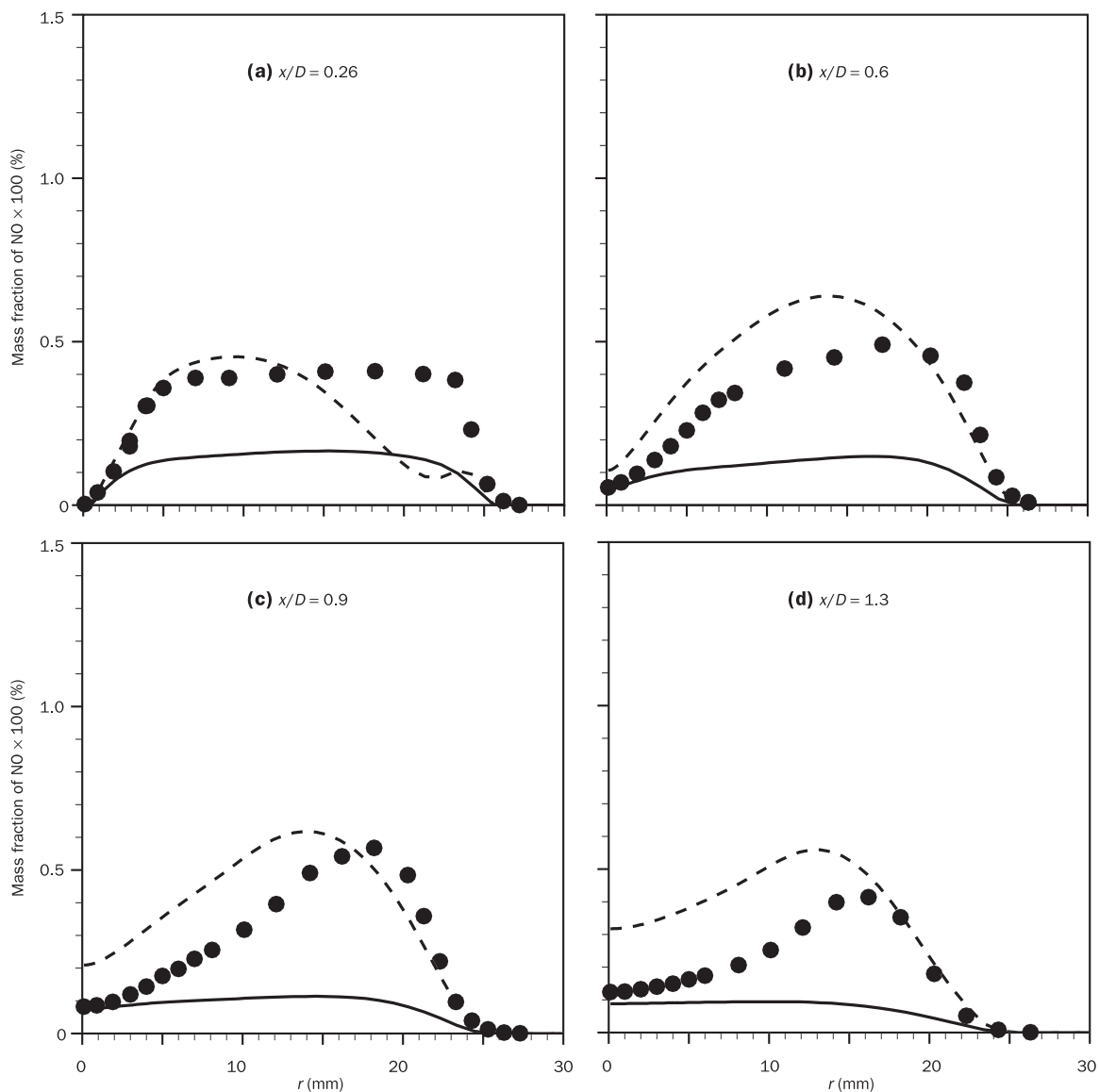



Figure 12.26 Mean NO mass fraction profiles: ●, measurements; ———, predictions (thermal NO only), - - - - - -, predictions including prompt NO_x

Further details of this example and the effects of using differential diffusion (non-unity Lewis numbers) can be found in Hossain and Malalasekera (2003) and Hossain (1999). The quality of the results presented in this example is very similar to that of Kim and Huh (2002), who used the conditional moment closure (CMC) method. Overall, laminar flamelet model predictions are very good, and the ability to incorporate detailed chemistry at an acceptable cost allows us to predict minor species and pollutants, which is a major advantage over simpler models such as SCRC.

12.31 Other models for non-premixed combustion

There are several other modelling concepts available for non-premixed combustion modelling. Among them, the conditional moment closure (CMC) model, pdf transport models and flame surface density models have been demonstrated to be successful in predicting turbulent combustion. It is beyond the scope of this chapter to describe these models in detail. The interested reader should consult the relevant literature. For the CMC model, Bilger (1993), Smith *et al*. (1992), Klimenko and Bilger (1999) and Kim and Huh (2002); for pdf transport models, Pope (1985, 1990, 1991) and Dopazo (1994); and for the flame surface density models, Marble and Broadwell (1977), Blunsdon *et al*. (1996), Beeri *et al*. (1996) and Veynante and Vervisch (2002) provide details and example applications. There is also growing interest in large eddy simulation (LES) models for turbulent combustion. Details can be found in Poinsot and Veynante, (2005), DesJardin and Frankel (1998), Cook and Riley (1998), Branley and Jones (2001) and Selle *et al*. (2004).

12.32 Modelling of premixed combustion

As mentioned in the introduction to this chapter, fuel and air are mixed prior to combustion in premixed combustion. The strength of the mixture may be expressed by the equivalence ratio. During combustion in a premixed flame, the flame front propagates at a certain speed and leaves burnt products behind the flame front. In premixed combustion, laminar and turbulent flame speeds and a parameter known as the reaction progress variable are used to formulate models. If we define T_u as the temperature of unburnt gas, T_b as the temperature of burnt gas and T as the flame temperature, then the reaction progress variable c is defined as

$$c = \frac{T - T_u}{T_b - T_u} \tag{12.203}$$

Sometimes the same reaction progress variable (c) is defined as

$$c = \frac{Y_F - Y_F^u}{Y_F^b - Y_F^u} \tag{12.204}$$

where Y_F, Y_F^u and Y_F^b are local, unburnt and burnt fuel mass fractions, respectively. With these definitions the value of the reaction progress variable is zero where the mixture is unburnt and unity where the mixture is fully burnt. It can be shown that, like mixture fraction in non-premixed combustion, the reaction progress variable c in premixed combustion is governed by the following transport equation (see Veynante and Vervisch, 2002):

$$\frac{\partial}{\partial t}\rho c + \frac{\partial}{\partial x_i}\rho u_i c = \frac{\partial}{\partial x_i}\left(\rho D \frac{\partial c}{\partial x_i}\right) + \dot{\omega} \tag{12.205}$$

Special variants of the main combustion models for diffusion flames (presumed-pdf, laminar flamelet, flame surface density, eddy break-up model etc.) have been formulated for premixed combustion. As we mentioned at the beginning of this chapter, we do not intend to describe the details of premixed combustion models here, and the interested reader is referred, among others, to Peters (1986), Veynante and Vervisch (2002) and Poinsot and Veynante (2005) for further details.

12.33 Summary

At the start of this chapter we introduced some basic thermodynamic and chemical kinetics concepts useful for the CFD combustion modeller. We have shown in detail how the fast chemistry assumption can be used to simplify combustion calculations, and demonstrated and its direct application to a laminar flame was demonstrated. The basics of chemical kinetics and reaction rates were introduced, and the advantage of using reduced mechanisms over detailed chemical mechanisms was discussed. The complexities arising from turbulent non-premixed combustion modelling were discussed, and the advantages of employing Favre-averaged equations were explained. The probability density function approach for the calculation of mean quantities in turbulent combusting flows was discussed in detail. We have also shown how the fast chemistry assumption can be used in conjunction with the pdf approach to calculate mean temperatures and species mass fractions. Some other approaches, such as the eddy break-up model, equilibrium models etc., were briefly discussed, and the need to incorporate some degree of detailed chemistry in order to be able to predict minor species was highlighted. Finally, we introduced the concepts of the laminar flamelet model, which allows the use of detailed chemistry. We discussed in some detail the methods for generating laminar flamelet libraries and calculating mean variables in the laminar flamelet method and showed how pollutant concentrations could be evaluated. A complete example illustrating the use of the laminar flamelet model was presented. Other models available for turbulent combustion were mentioned, along with a brief description of the main features of models for premixed combustion. This chapter should be useful for novice combustion modellers wishing to gain some overall understanding of the basic approach used in CFD. In the interest of brevity many intermediate details have been omitted in the presentation of various concepts. The reader should follow up appropriate references and gather full details in order to understand the finer points of various models.

Chapter thirteen Numerical calculation of radiative heat transfer

13.1 Introduction

Conduction, convection and radiation are the three modes of heat transfer. In previous chapters we have seen how problems involving multi-dimensional conductive and convective heat transfer can be solved using CFD. Conduction-only problems are simply dealt with by solving the heat conduction equation, i.e. diffusion equation (4.1). When fluid flow is involved the resulting convective heat transfer problem is solved by tackling the enthalpy equation (2.27) alongside the Navier–Stokes equations (2.32a–c) and the continuity equation (2.4). The boundary conditions for the enthalpy equation take care of heat transfer into and out of the computational domain across its boundaries. The internal distribution of heat source and sink processes and the transport of heat by means of diffusion and convection determine the enthalpy distribution due to fluid flow.

The third mechanism of heat transfer – **thermal radiation** – is caused by energy emission in the form of **electromagnetic waves** or **streams of photons**. Radiative energy sources emit a broad-band spectral distribution with maximum energy content at a wavelength that is determined by the source temperature. Most engineering systems emit thermal radiation at infrared wavelengths (0.7–100 μm). The peak wavelength of sources at room temperature will be around 10 μm, whereas an increasing fraction of the radiation will be emitted at visible wavelengths when the source temperature exceeds 1000–1500 K.

Engineering problems with significant radiation effects

Radiative heat transfer is often neglected in CFD calculations, because the majority of engineering problems are dominated by high rates of convective heat transfer. There are, however, several practically important categories of problems where radiative heat transfer should be considered. Three prominent examples are:

1 Manufacturing processes involving lasers or other high-energy beams: these involve exposure of materials to radiation with very high energy density. The radiative heat input will dominate these problems.
2 Combustion equipment: there is usually vigorous convection in combustors, but the chemical reactions generate operating temperatures that are sufficiently high for radiative heat fluxes to be of similar order of magnitude.
3 Naturally ventilated spaces in buildings: the temperatures are low so radiative heat fluxes are modest but comparable in size with convective heat fluxes since buoyancy-driven flow velocities are often small.

All these problems involve combined-mode or conjugate heat transfer: coupled radiation and conductive heat transfer or, if fluid flow occurs, coupled radiation, conductive and convective heat transfer. Laser-based material processing falls within the complex category of free boundary problems with phase change. We do not discuss these further. In this section we focus on coupled radiation/CFD problems in enclosures or flow domains with fixed boundaries. Radiation acts to redistribute energy through interactions within the fluid, through radiative exchanges between boundary surfaces, and through interactions between the surfaces and the fluid.

Definitions

We give a selection of basic facts and definitions of radiative heat transfer needed to support our review of the main solution algorithms.

The **emitted radiative heat flux** is a strong function of the temperature of the substance. Materials at higher temperature will emit more radiation. The rate of heat flow per unit **surface** area emitted by a radiating surface is called its **emissive power** E (units W/m^2). For a so-called black body the emissive power is related to its temperature by $E_b = \sigma T^4$, where $\sigma = 5.67 \times 10^{-8}$ ($W/m^2.K^4$) is Stefan–Boltzmann's constant and T is absolute temperature (K).

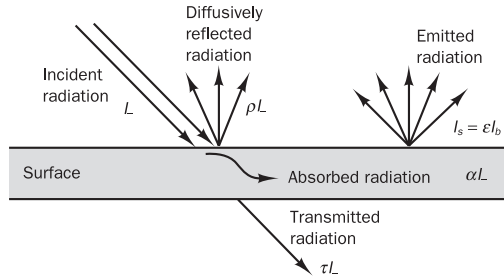
The **incident radiative heat flux** at a certain location varies as a function of orientation of the receiver relative to the radiation source. The **intensity** I (units $W/m^2.sr$) is the rate of heat flow received per unit area *perpendicular to the rays* and per unit solid angle (steradians = sr), and is a quantity with a magnitude that will vary with direction.

The black-body emissive power $E_b = \sigma T^4$ can also be expressed as a (non-directional) **black-body intensity** using $I_b = E_b/\pi = \sigma T^4/\pi$. The black-body intensity is used to compute the emitted intensity from surfaces and fluids.

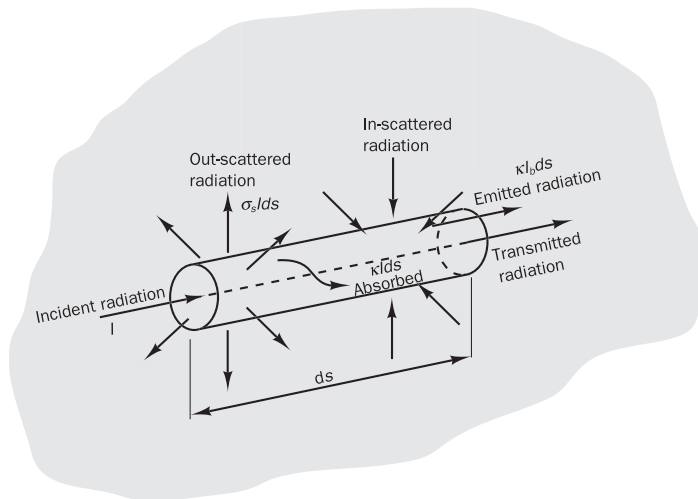
Surface properties

The emissive power of an ideal black surface is, of course, given by $E_s = \sigma T_s^4$, where T_s is the surface temperature. Real surfaces usually emit less radiative heat. The ratio of the heat flux emitted by a real surface and a black surface at the same temperature is called the **surface emissivity** ε. Hence, the emissive power of a real surface with surface emissivity ε is given by $E_s = \varepsilon \sigma T_s^4$. The (non-directional) **emitted intensity** I_s of a real surface is the product of its surface emissivity and the black-body intensity: $I_s = \varepsilon I_b = \varepsilon \sigma T_s^4/\pi$.

Figure 13.1 shows that surfaces interact with incident radiation in three ways. The incident radiation can be (i) absorbed, (ii) reflected or (iii) transmitted. The fraction **absorbed** is denoted by α, the fraction **reflected** by ρ and the fraction **transmitted** by τ. Of course, the sum of the absorbed, reflected and transmitted fractions equals unity: $\alpha + \rho + \tau = 1$. In many coupled CFD/radiation problems the surfaces will be (i) **opaque**, so $\tau = 0$, and (ii) **diffusely reflecting**, i.e. the reflected radiation leaves the surface in all directions irrespective of the incident angle of the incoming radiation. Real surfaces may have different properties, e.g. non-zero transmissivity τ or specular reflection (equal angle between surface normal and incident and reflected radiation). All the surface properties are dependent on the type of material, the surface roughness and the presence of surface contaminants, as well as the temperature, and surface properties may also depend on the direction and wavelength of the incident radiation.

Figure 13.1 Radiation processes at a surface

Fluid properties

We have noted before that air is transparent to the infrared radiation typically associated with room temperature. The species present in clean atmospheric air do not absorb readily at these wavelengths, and this **fluid medium** will not participate in the radiative heat exchanges. Products of combustion reactions, on the other hand, contain large amounts of carbon dioxide and water vapour, which are both strong absorbers/emitters in the infrared part of the spectrum. Moreover, scattering may occur under certain conditions if a combustion reaction produces particulate matter in the form of soot, or if the fuel is in the form of solid particles. Thus, in combusting flows incident radiation is absorbed, transmitted and/or scattered and radiation may be emitted (see Figure 13.2). In such cases the fluid is termed a **participating medium**.

Figure 13.2 Schematic of a pencil of ray in a medium subjected to emission, absorption, scattering and transmission processes.

The strength of the interactions between a participating fluid medium and radiation can be measured in terms of its **absorption coefficient** κ and its **scattering coefficient** σ_s, both of which have units m^{-1}. The sum of these two properties is called the **extinction coefficient** $\beta = \kappa + \sigma_s$. The **emitted intensity** I_f of a participating fluid medium is the product of the absorption coefficient and the black-body intensity: $I_f = \kappa I_b = \kappa \sigma T_f^4/\pi$,

where T_f is the fluid temperature. The distribution of emitted intensity by a point radiation source in a participating fluid medium is uniform in all directions, but the scattered intensity is generally not. The distribution of the latter can described by the so-called **scattering phase function** $\Phi(\mathbf{s}_i, \mathbf{s})$, which is defined as the fraction of radiation incident upon the medium along direction vector $\mathbf{s}_i$, which is subsequently scattered in direction $\mathbf{s}$. The absorption coefficient, scattering coefficient and the scattering phase function are properties of the participating medium. The absorption coefficient κ will depend on the fluid temperature and species concentrations and may also depend on pressure. The scattering coefficient σ_s and the scattering phase function $\Phi(\mathbf{s}_i, \mathbf{s})$ depend on the size, concentration, shape and material characteristics of suspended particulate matter in the fluid. All these radiative properties generally have a complex wavelength dependence.

The nature of coupled CFD/radiation problems

Radiation can be considered as an electromagnetic wave phenomenon. Therefore, the propagation speed of thermal radiation is the speed of light, which, at 3×10^8 m/s, is at least 10^5 times as fast as the speed of sound of common fluids encountered in engineering problems. This large separation of the velocity scales means that radiative heat exchanges are always in quasi-steady state. This means that radiation propagates sufficiently fast to effectively adjust itself immediately to variations in flow conditions and/or boundary conditions.

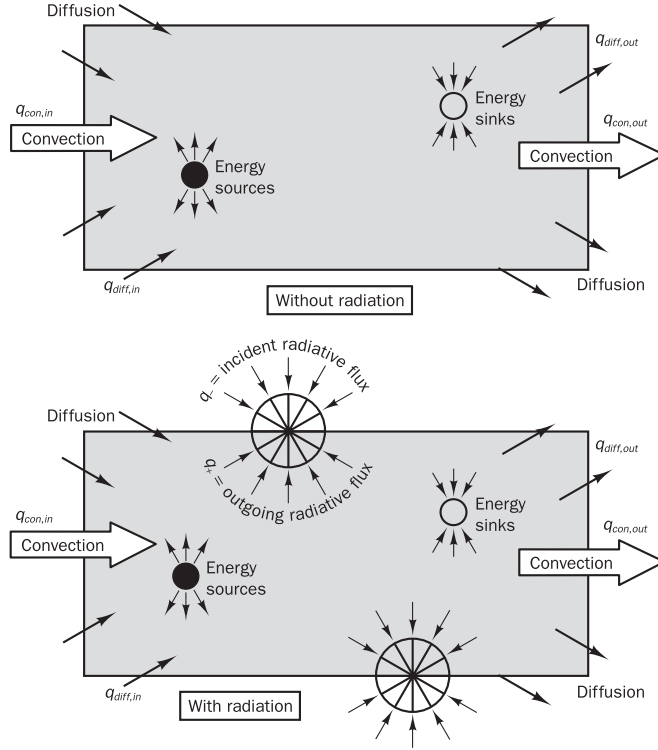
There is no direct coupling between radiation and the flow field, since radiation or radiation properties of fluids and boundaries do not depend directly on the fluid velocity. However, the flow field influences the spatial distributions of temperature and species concentration. These determine the intensity of radiation emitted by boundary surfaces and participating fluids as well as the radiation properties of a participating medium. This ensures that there is strong indirect coupling between the flow field and radiation environment, which is particularly important in combustion problems where product species are at high temperature. We now review the consequences for CFD of these coupling effects:

1. If the fluid is absorbing/emitting and/or scattering there will be an additional radiation source term in the energy equation
2. Radiation effects will cause changes in the boundary conditions of the energy equation

Radiation source term in energy equation

In a steady flow without radiation there is a balance within each fluid control volume between the sources and sinks of energy on the one hand and diffusive and convective fluxes across the boundaries on the other hand (see Figure 13.3). Depending on the radiative properties of the fluid inside the control volume, it may absorb, emit and/or scatter radiation. When these effects are significant there will be additional heat fluxes across the control volume boundaries due to radiation. The net radiative heat flux into or out of the control volume will appear in the energy equation as a source or sink, respectively.

Thus, the steady flow energy balance for the control volume in the two cases is as follows:

Figure 13.3 Heat fluxes and sources for a fluid control volume without and with radiation

Without radiation: $$\int_{CV} S_h \mathrm{d}V = \int_A \mathbf{q}_{conv} \,.\, \mathbf{n}\mathrm{d}A + \int_A \mathbf{q}_{diff} \,.\, \mathbf{n}\mathrm{d}A \tag{13.1a}$$

Net energy source in CV	=	Net convective flux across boundaries	+	Net diffusive flux across boundaries

With radiation: $$\int_{CV} S_h \mathrm{d}V + \int_{CV} S_{h,rad} \mathrm{d}V = \int_A \mathbf{q}_{conv} \,.\, \mathbf{n}\mathrm{d}A + \int_A \boldsymbol{q}_{diff} \,.\, \mathbf{n}\mathrm{d}A + \int_A \boldsymbol{q}_{rad} \,.\, \mathbf{n}\mathrm{d}A \tag{13.1b}$$

Non-radiative energy source in CV	+	Net radiative energy source in CV	=	Net convective flux	+	Net diffusive flux	+	Net radiative flux across boundaries

So

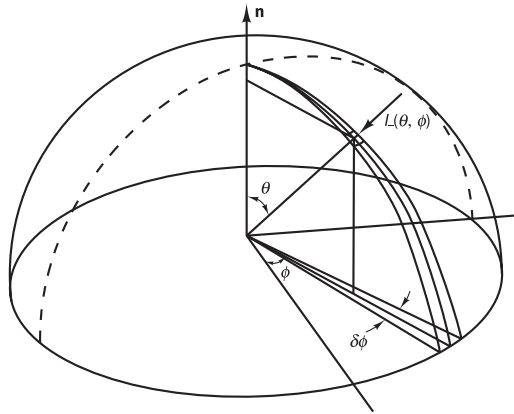
$$\bar{S}_{h,rad} = \frac{1}{\Delta V} \int_A \mathbf{q}_{rad} \,.\, \mathbf{n}\mathrm{d}A = \frac{1}{\Delta V} \int_A (q_- - q_+)\mathrm{d}A \tag{13.1c}$$

where $\bar{S}_{h,rad}$ is the net radiative source per unit volume for the energy equation.

In (13.1c) the net radiative heat flux $\mathbf{q}_{rad} \cdot \mathbf{n}$ at a point on the control volume boundary has been written as the difference between the **incident radiative heat flux** q_- and the **outgoing heat flux** q_+. The integral of the difference $q_- - q_+$ over the bounding surface of the control volume gives the net radiative heat flux into the control volume to obtain the net radiative heat source for the energy equation. To obtain the incident heat flux q_- at a point it is necessary to integrate the intensity, which varies with direction, and hence we must consider *all possible incoming ray directions*. It can be shown (Modest, 2003) that this can be written as an integration over a unit hemisphere (i.e. over a solid angle of 2π steradians) surrounding the point *just outside* the control volume boundary (indicated as a semi-circle in Figure 13.3). So,

$$q_- = \int_{2\pi} I_-(\mathbf{s})\, \mathbf{s} \cdot \mathbf{n}\, d\Omega = \int_0^{2\pi} \int_0^{\pi/2} I_-(\theta, \phi) \cos(\theta) \sin(\theta)\, d\theta\, d\phi \quad (13.2)$$

The dot product $\mathbf{s} \cdot \mathbf{n}$ in the first integral ensures that we consider the component of the incident radiative heat flux vector in the direction of the outward surface normal $\mathbf{n}$. We note that the swap of argument of the incident intensity I_- from ray direction vector $\mathbf{s}$ to angular co-ordinates (θ, ϕ) is simply a matter of notation. The co-ordinate system is illustrated in Figure 13.4.

Figure 13.4 Angular notation for equation (13.2), incoming radiative flux

To obtain the outgoing radiative heat flux q_+ we integrate over all possible ray directions pointing outwards at the point. This corresponds to an integration over a unit hemisphere (indicated as a semi-circle in Figure 13.3) surrounding the point just inside the control volume boundary. So,

$$q_+ = \int_{2\pi} I_+(\mathbf{s})\, \mathbf{s} \cdot \mathbf{n}\, d\Omega = \int_0^{2\pi} \int_{\pi/2}^{\pi} I_+(\theta, \phi) \cos(\theta) \sin(\theta)\, d\theta\, d\phi \quad (13.3)$$

$I_+(\mathbf{s})$ and $I_+(\theta, \phi)$ represent the outgoing intensity of radiation in the direction of vector $\mathbf{s}$ corresponding to angular direction (θ, ϕ). The limits in the

θ-integration have been adjusted to indicate the unit hemisphere just inside the control volume with outward-pointing radiation rays.

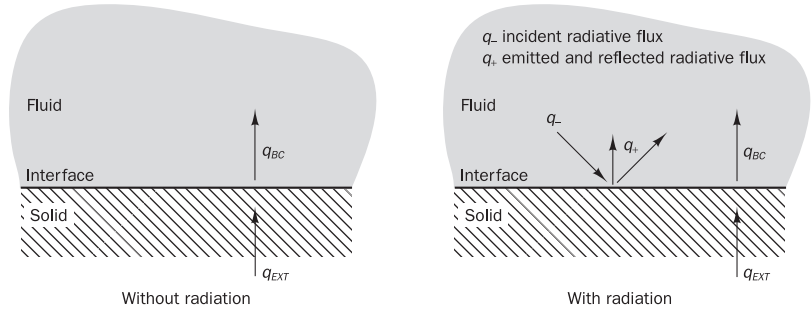
The effect of radiation on boundary conditions

Without radiation there is a balance between the conductive and/or convective heat fluxes at the interface that constitutes the imaginary boundary between two materials. Radiation will cause an additional heat flux towards the interface due to incident radiation and an extra outgoing heat flux associated with emission of radiation.

The overall heat balance for the interface between the fluid and the boundary states that the heat flux out of the interface must be equal to the heat flux into the interface. Thus, as illustrated in Figure 13.5,

Without radiation: $q_{BC} = q_{EXT}$ (13.4a)

With radiation: $q_{BC} + q_+ = q_{EXT} + q_- \Rightarrow q_{BC} = q_{EXT} + (q_- - q_+)$ (13.4b)

Figure 13.5 Boundary conditions with and without radiation

Both the **incident radiative heat flux** q_- and the **outgoing radiative heat flux** q_+, which is the sum of emitted and reflected heat fluxes, depend on the intensity of the radiation. To obtain the incident heat flux at a point on the surface we use equation (13.2) with the surface normal **n** pointing into the fluid region.

The **outgoing radiative heat flux** is given by

$$q_+ = \varepsilon E_s + (1 - \varepsilon)q_- = \varepsilon\sigma T_s^4 + (1 - \varepsilon)q_- \tag{13.5}$$

The first term on the right hand side is the emitted heat flux and the second term is the reflected heat flux. The latter is just equal to the fraction $(1 - \varepsilon)$ times the incident heat flux q_- in the case of diffuse reflection. The corresponding non-directional intensity of the outgoing radiation is

$$I_+ = q_+/\pi = [\varepsilon\sigma T_s^4 + (1 - \varepsilon)q_-]/\pi \tag{13.6}$$

If we use a fixed temperature boundary condition, we need to bear in mind that the net radiative heat flux $q_- - q_+$ will vary across the boundary surfaces. Radiation problems always involve combined-mode heat transfer, and surface locations experiencing positive values of $q_- - q_+$ are likely to heat up compared with areas with zero or negative $q_- - q_+$. Similarly, if we use a heat flux boundary condition, a CFD computation requires us to specify the heat flux into the fluid q_{BC}, but it is generally more convenient to specify the external heat flux q_{EXT}. As we can see in the interface heat balance, $q_{BC} = q_{EXT}$ for a

flow without radiation. In an environment with radiation, however, the heat flux going into the fluid q_{BC} at a boundary point will be different from the external heat flux q_{EXT} by an amount equal to the local net radiative heat flux $q_- - q_+$. These remarks serve to emphasise that completely accurate specification of surface boundary conditions is difficult without a priori knowledge of the radiation environment.

13.2 Governing equations of radiative heat transfer

The incident intensity I_- and the outgoing intensity I_+, which are needed in equations (13.2) and (13.3) for the radiative heat fluxes, need to be computed from the governing equation of radiative transfer. The general relationship that governs the changes in intensity at a point along a radiation ray due to emission, absorption and scattering in a fluid medium is as follows (Modest, 2003):

$$\frac{dI(\mathbf{r}, \mathbf{s})}{ds} = \kappa I_b(\mathbf{r}) - \kappa I(\mathbf{r}, \mathbf{s}) - \sigma_s I(\mathbf{r}, \mathbf{s}) + \frac{\sigma_s}{4\pi} \int_{4\pi} I_-(\mathbf{s}_i)\, \Phi(\mathbf{s}_i, \mathbf{s})\, d\Omega_i \qquad (13.7)$$

Rate of change of intensity per unit path length	=	Emitted intensity	−	Absorbed intensity	−	Out-scattered intensity	+	In-scattered intensity

where $I(\mathbf{r}, \mathbf{s})$ is the radiation intensity at a given location indicated by position vector $\mathbf{r}$, in the direction $\mathbf{s}$ within a small pencil of rays, travelling through a participating medium (see Figure 13.2 and Modest, 2003). The in-scattering integral on the right hand side of equation (13.7) accounts for the effect of intensity $I_-(\mathbf{s}_i)$ incident upon the point at $\mathbf{r}$ from all possible directions $\mathbf{s}_i$. This involves integration over a unit sphere (i.e. a solid angle of 4π steradians) surrounding the point in the medium.

The absorbed and out-scattered intensity are often drawn together by defining the extinction coefficient $\beta = \kappa + \sigma_s$ as the sum of the absorption coefficient and the scattering coefficient. Thus,

$$\boxed{\frac{dI(\mathbf{r}, \mathbf{s})}{ds} = \kappa I_b(\mathbf{r}) - \beta I(\mathbf{r}, \mathbf{s}) + \frac{\sigma_s}{4\pi} \int_{4\pi} I_-(\mathbf{s}_i)\, \Phi(\mathbf{s}_i, \mathbf{s})\, d\Omega_i} \qquad (13.8)$$

This is the radiative transfer equation (RTE), which governs radiative heat transfer along a ray path indicated by the direction vector $\mathbf{s}$. If the incident intensity field for the in-scattering integral (last term on the right hand side of the RTE) is somehow known, the RTE is a first-order ordinary differential equation.

Many computational algorithms for radiative heat transfer proceed to simplify the radiative transfer equation by introducing non-dimensional optical co-ordinates:

$$\tau = \int_0^s (\kappa + \sigma_s)\, ds' = \int_0^s \beta\, ds' \qquad (13.9)$$

and the single scattering albedo ω defined as

$$\omega = \frac{\sigma_s}{\kappa + \sigma_s} = \frac{\sigma_s}{\beta} \tag{13.10}$$

The radiative transfer equation can be re-written as

$$\frac{dI(\tau, \mathbf{s})}{d\tau} = -I(\tau, \mathbf{s}) + (1 - \omega)I_b(\tau) + \frac{\omega}{4\pi}\int_{4\pi} I_{-}(\mathbf{s}_i)\,\Phi(\mathbf{s}_i, \mathbf{s})\,d\Omega_i \tag{13.11}$$

Finally, we introduce the source function, which is defined as follows:

$$\boxed{S(\tau, \mathbf{s}) = (1 - \omega)I_b(\tau) + \frac{\omega}{4\pi}\int_{4\pi} I_{-}(\mathbf{s}_i)\,\Phi(\mathbf{s}_i, \mathbf{s})\,d\Omega_i} \tag{13.12}$$

This enables us to write the radiative transfer equation in a form with a very simple appearance:

$$\boxed{\frac{dI(\tau, \mathbf{s})}{d\tau} + I(\tau, \mathbf{s}) = S(\tau, \mathbf{s})} \tag{13.13}$$

Boundary conditions

Equation (13.8) or (13.13) is solved by integration along ray paths starting at the boundaries of the computational domain, which requires initial conditions on these surfaces. For a point on a diffusely emitting and reflecting opaque surface located at position vector $\mathbf{r}_w$ we use expression (13.6) to obtain the initial condition of a ray propagating along direction vector $\mathbf{s}$:

$$\boxed{I(\mathbf{r}_w, \mathbf{s}) = \varepsilon I_b(\mathbf{r}_w) + \frac{(1-\varepsilon)}{2\pi}\int_{2\pi} I_{-}(\mathbf{r}_w, \mathbf{s}_i)\mathbf{n}\,.\,\mathbf{s}_i\,d\Omega_i} \tag{13.14}$$

This is the most widely used boundary condition in combustion-related radiative transfer problems. Expressions for other possible boundary conditions can be found in Modest (2003).

Incident intensity integrals

The RTE (13.8) and boundary condition (13.14) contain integrals of intensity over a sphere and hemisphere, respectively. To evaluate these it is necessary to know the integrands beforehand as a function of the direction of the incident radiation. Since the incident intensity integrals are unknown and must be solved alongside the intensity along an RTE ray path, equation (13.8) is not an ordinary differential equation but actually an integro-differential equation. This type of equation is clearly very different in nature from the transport equations we have seen before. This requires solution of RTEs along all the relevant ray paths, so the transport of heat by radiation is always three-dimensional. Moreover, radiative transfer is heavily dependent on the

geometry. Therefore, the calculation of radiative heat transfer requires an approach that is quite different from the finite volume method we have discussed elsewhere in this book. We will discuss the most prominent general-purpose solution algorithms in the next section.

13.3 Solution methods

Exact analytical solutions of the RTE are not available except in a small number of idealised cases. In most practical problems further simplification of the transfer equation is not possible. In particular, the boundary conditions are dictated by the problem geometry, and full three-dimensional effects and all angular directions must be considered in solving the transfer equation. Before we discuss the details of solution methods we examine the RTE and boundary conditions to gain insight into some of the general features of likely solution procedures. First, we note that the incident intensity integrals in (13.2) and (13.8) are unknown at the start of a calculation, so an iterative approach is required. Assumed values of the surface intensities are initially used and, after solving the RTE for a sufficiently large number of ray paths, the incident intensity integrals can be evaluated. This enables us to make improved estimates of the boundary conditions and the in-scattering integrals to carry out another round of RTE solutions. This process is iterated until there is no further change in the solution.

Further complications arise in combusting systems, because the equation of transfer contains terms involving the unknown temperature field and radiative properties, which are dependent on temperature and composition. Combustion effects dominate the temperature field, and the radiative properties of the medium should be determined from the concentration of species of combustion. The interaction between radiative heat transfer and flow enters into the fluid flow calculation through the source term (see equation (12.80) in Chapter 12) and wall heat transfer effects. Due to this coupling of temperature and radiative properties of the medium the solution of the RTE in combustion problems requires an outer iteration loop, which is executed until the solution satisfies all the fluid flow equations and the RTE.

Finally, the radiation properties of combustion products are dependent on the wavelength of the radiation and, for accurate simulations, it is necessary to perform spectrally resolved radiation calculations or to model the effect of this wavelength dependence.

The calculation of radiation is numerically challenging and resource intensive, since algorithms have to compute radiation intensity as a function of position (x, y, z) in the computational domain, angular direction (θ, ϕ) and, in the most accurate calculations, radiation wavelength λ. In our brief review of techniques for radiation calculations we focus our attention on computational methods for the dependence of radiation intensity on position and direction and illustrate their application by means of three examples of increasing complexity. We finish the chapter by making some brief remarks on issues relating to wavelength-dependent radiation calculations.

Over the years many methods have been developed for the solution of radiative heat transfer. These include various analytical approximation techniques and a suite of numerical methods. Some early methods have now largely been abandoned because of their limited applicability to general situations. Other methods, such as the zone method, P–N methods, flux methods and finite element methods are not discussed here. Details can be found in the reviews and texts by Sarofim (1986), Viskanta and Mengüç

(1987), Howell (1988), Siegel and Howell (2002), Modest (2003), Carvalho and Farias (1998) and Maruyama and Guo (2000). In this chapter we discuss four of the most popular general-purpose radiation algorithms:

- Monte Carlo method (Howell and Perlmutter, 1964)
- Discrete transfer method (Lockwood and Shah, 1981)
- Discrete ordinate method (Chandrasekhar, 1960; Hyde and Truelove, 1977; Fiveland, 1982, 1988)
- Finite volume method (Chui *et al,* 1992, Chui and Raithby, 1993)

These methods each have a different way of treating the angular dependence and spatial variation of intensity. The Monte Carlo and discrete transfer methods are based on ray tracing. The last two methods use numerical discretisation of the distance and directional integrals, so there is not such an obvious connection with rays.

13.4 Four popular radiation calculation techniques suitable for CFD

13.4.1 The Monte Carlo method

In the Monte Carlo (MC) ray tracing method the radiative heat transfer is calculated by randomly releasing a statistically large number of energy bundles and tracking their progress from their emission points through the medium. The domain boundary and the medium, which can be emitting, absorbing and scattering, are usually discretised into surface and volume elements for calculation purposes. The method is independent of the co-ordinate system and therefore applicable to arbitrarily shaped and complex configurations. Several variations of Monte Carlo algorithms are available, and are described in Farmer (1995) and Howell (1998). Depending on the chosen variation of the Monte Carlo method, the emission points can be boundary surface elements or volume elements within the media. The energy of an individual bundle is taken as the total emissive power of the originating sub-region divided by the number of bundles (N) released from this area. The method requires ray tracing to compute the path followed by the bundles through the computational domain. Each bundle can gain or lose energy along its path, depending on the properties of the medium. The amount of energy gained or lost in this transfer process is used to calculate the net energy source or sink in the medium due to radiation. It will eventually strike another surface, the properties of which are used to determine whether the bundle is absorbed by the surface element. The method can accommodate all properties of radiative transfer including non-isotropic scattering, spectral effects and complex surface properties.

There are several probabilistic features, which give the Monte Carlo method its name. Random numbers are used to determine the emission location and direction of the energy bundles as well as the fractions emitted, absorbed and scattered during interactions with the medium and boundary surfaces. Random number generators provided by present-day programming language compilers or a dedicated random number generator algorithm may be used to draw random numbers between 0 and 1. The calculation process starts by drawing a set of independent random numbers R_x, R_y, R_z to determine the location of the emission. For a two-dimensional rectangular surface element a location on the element may be identified as

$$x = x_o + R_x \Delta x \tag{13.15a}$$

$$y = y_o + R_x \Delta y \tag{13.15b}$$

where x_o, y_o are the co-ordinates of the vertex of the element, and Δx and Δy are dimensions of the cell measured from x_o, y_o. The polar and azimuthal angles are used to determine the direction of a bundle. For a diffuse emitter, for example, the angles are given by

$$\theta = \sin^{-1}\sqrt{R_\theta} \tag{13.16}$$

$$\phi = 2\pi R_\phi \tag{13.17}$$

where R_θ and R_ϕ are random numbers. As stated above, ray tracing is used to calculate the path followed by each energy bundle.

For a bundle with initial energy E, travelling through a grey medium with a constant absorption coefficient κ, the energy remaining in the bundle (E_{bundle}) and energy gained by the medium ($E_{absorbed}$) are given by

$$E_{bundle} = E(e^{-\kappa s}) \tag{13.18}$$

$$E_{absorbed} = E(1 - e^{-\kappa s}) \tag{13.19}$$

where s is the path length travelled, which is obtained from

$$s = -\frac{1}{\kappa}\ln(R_l) \tag{13.20}$$

where R_1 is a random number (see Siegel and Howell, 2002).

When an energy bundle strikes a surface its absorptivity (α) is interpreted as the probability that the bundle is absorbed. A random number R_α is compared with the absorptivity value α to determine the process:

$$R_\alpha < \alpha \quad \text{the bundle is absorbed} \tag{13.21}$$

$$R_\alpha > \alpha \quad \text{the bundle is reflected} \tag{13.22}$$

By keeping track of energy absorbed or emitted within volumes and surface elements, appropriate radiation quantities such as surface heat flux, divergence or temperature may be evaluated depending on the problem. The total energy absorbed ($E_{absorbed}$) by surfaces and volumes is tallied during the simulation procedure. Then the radiation source term and surface heat flux may be evaluated from

$$\boxed{S_{rad,h_f} = \frac{1}{V_j}\left(\sum^{\text{absorbed bundles}} E_{absorbed} - \sum^{\text{emitted bundles}} E_{emitted}\right)} \tag{13.23}$$

$$\boxed{Q_{si} = \frac{1}{A_i}\left(\sum^{\text{absorbed bundles}} E_{absorbed} - \sum^{\text{emitted bundles}} E_{emitted}\right)} \tag{13.24}$$

In problems with prescribed wall temperature the initial intensity is unknown; therefore simulation has to continue until convergence is achieved for the heat flux value.

The accuracy of the Monte Carlo method is proportional to the square root of the total number of bundles released in the calculations. Since substantial probabilistic elements are involved, the standard error in the mean values of calculated quantities (e.g. flux or cell source term) can be estimated.

This gives a unique – amongst radiation methods – indication of the uncertainty of the results. The origin of the expressions in this section can be found in Siegel and Howell (2002), Modest (2003) and Mahan (2002). Details of the treatment of more advanced problems involving spectral and angular dependent properties and/or scattering can be found in the above textbooks and also in Farmer (1995).

13.4.2 The discrete transfer method

In the discrete transfer method (DTM) of Lockwood and Shah (1981), the solution proceeds by first discretising the radiation space into homogeneous surface and volume elements. Rays are emitted from the centre of each boundary surface element, with position vector **r**, in directions determined by discretising the 2π hemispherical solid angle above the surface into finite solid angles, $\delta\Omega$. Shah (1979) chose to divide the hemisphere into N_θ equal polar angles and N_ϕ equal azimuthal angles such that $N_\Omega = N_\theta N_\phi$ and

$$\delta\theta = \frac{\pi}{2N_\theta} \quad \delta\phi = \frac{2\pi}{N_\phi} \tag{13.25}$$

These features are illustrated in Figures 13.6 and 13.7.

In vector terms a ray is traced through the centre of each solid angle element, in the direction $-\mathbf{s}_k$, until it strikes a boundary at $\mathbf{r}_L = \mathbf{r} - \mathbf{s}_k L_k$, such that the ray path length is $L_k = |\mathbf{r} - \mathbf{r}_L|$. To calculate the contribution due to the solid angle element to the incident intensity at the origin, the ray is followed back, starting at $\mathbf{r}_L$, to origin $\mathbf{r}$. The intensity distribution along its path is solved with the recurrence relation

$$I_{n+1} = I_n e^{-\beta\delta s} + S(1 - e^{-\beta\delta s}) \tag{13.26}$$

where n and $n+1$ designate successive boundary locations, separated by a distance δs, as the ray passes through each medium control volume. The source function S includes the scattering integral in its angular discretised form:

$$S = (1-\omega)I_b + \frac{\omega}{4\pi}\int_{4\pi} I_-(\mathbf{s}_i)\,\Phi(\mathbf{s}, \mathbf{s}_i)\,d\Omega_i$$

$$= (1-\omega)I_b + \frac{\omega}{4\pi}\sum_{i=1}^{N} I_{-,ave}(\mathbf{s}_i)\,\Phi(\mathbf{s}, \mathbf{s}_i)\,\delta\Omega_i \tag{13.27}$$

where the averaged intensity $I_{-,ave}(\mathbf{s}_i)$ is taken as the arithmetic mean of the entering and leaving radiant intensities for each ray passing through the cell volume within the finite solid angle $\delta\Omega$. The finite solid angle is evaluated for each angular sub-division as

$$\delta\Omega = \int_{\delta\phi}\int_{\delta\theta} \sin\theta\, d\phi\, d\theta = 2\sin\theta\sin(\delta\theta/2)\delta\phi \tag{13.28}$$

Source function S is assumed to be constant over the interval. A cell-averaged (directionally independent) value is taken for S. This considerably simplifies the analysis, but as a consequence of this approximation the DTM does not have the ability to describe scattering anisotropy.

Figure 13.6 Angular discretisation and representative ray selection in the discrete transfer method: (a) angular discretisation in the azimuthal direction; (b) angular discretisation in the polar direction; (c) selection of ray directions for a single dϕ angular sector

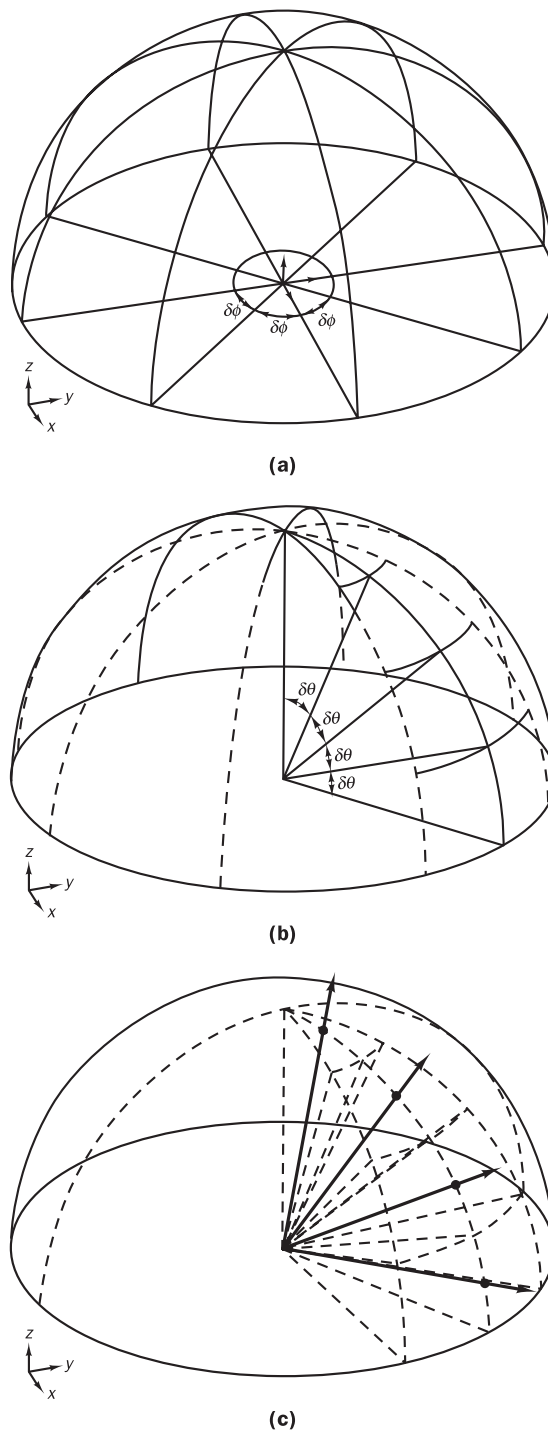
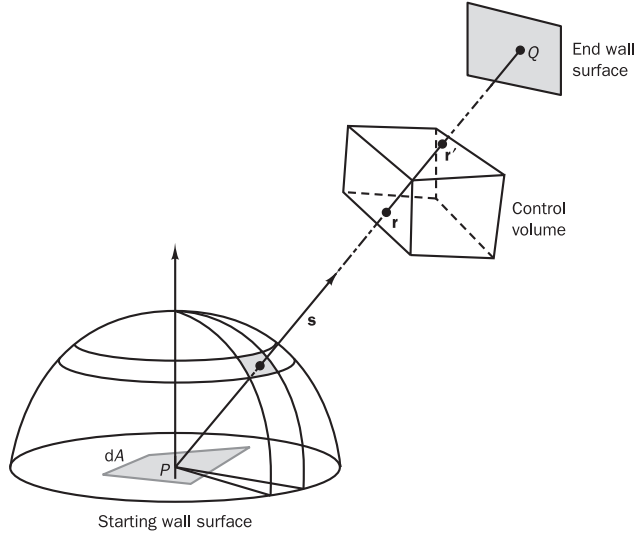



Figure 13.7 An illustration of the discrete transfer method

The initial intensity value of each ray is evaluated at the originating surface element and is given by

$$I_o = q_+/\pi \tag{13.29}$$

with

$$q_+ = \varepsilon E_s + (1 - \varepsilon)q_- \tag{13.30}$$

Equation (13.26) is applied in the direction towards the origin of each ray, and the incident radiative heat flux q_- is evaluated by summing contributions over all solid angles, assuming that the intensity is constant over each finite solid angle. This gives

$$q_- = \sum_{N_\Omega} I_-(\mathbf{s})\ \mathbf{s}\ .\ \mathbf{n}\delta\Omega = \sum_{N_R} I_-(\theta, \phi)\cos(\theta)\sin(\theta)\sin(\delta\theta)\delta\phi \tag{13.31}$$

where N_R is the number of incident rays arriving at the surface element.

Since q_+ in (13.30) depends on the value of q_-, an iterative solution is required, unless the surfaces are black ($\varepsilon = 1$). In an iterative calculation the initial intensity leaving a surface is evaluated on the basis of its own temperature only. Estimates of the incident radiative heat flux will be available after the first iteration to calculate corrected intensities leaving boundary surfaces using (13.29) and (13.30). The process is repeated until the difference between successive values of the negative flux is within a specified limit.

When the calculation has converged the net radiative heat flow out of each surface element with area A_i is computed from

$$\boxed{Q_{si} = A_i(q_+ - q_-)} \tag{13.32}$$

DTM calculates the radiative source for each medium element via an energy balance. Lockwood and Shah (1981) evaluated the radiative source associated with the passage of each ray through a volume n by means of

$$\delta Q_{gk} = \int_{\delta\Omega} (I_{n+1} - I_n) A_s (-\mathbf{s}_k \cdot \mathbf{n}) \mathrm{d}\Omega_k$$

$$= (I_{n+1} - I_n) A_s \cos\theta_k \sin\theta_k \sin(\delta\theta_k)\delta\phi_k \tag{13.33}$$

where A_s is the area of the surface element from which the ray was emitted. Summing the individual source contributions from all the N rays passing through a volume element, and then dividing this value by its volume, ΔV, gives the divergence of radiative heat flux as

$$\boxed{S_{h,rad} = \nabla \cdot \mathbf{q}_r = \frac{1}{\Delta V}\sum_{k=1}^{N} \delta Q_{g,k}} \tag{13.34}$$

Each DTM ray originates from a certain surface element and selects its initial intensity on the basis of the properties of this element. However, the incremental solid angle, represented by the ray, may partly cover an adjacent surface element. Since the influence of the initial intensity decays exponentially along its transmission path, the resulting inaccuracy is likely to be small.

The accuracy of the DTM depends on two factors, surface discretisation (number of surface elements used to fire rays) and angular discretisation (number of rays used per point). Mathematical expressions for the errors resulting from these discretisation practices have been derived for simple situations (Versteeg *et al*, 1999a, b). The studies using transparent media show that the decay rate of the angular discretisation error ε_H with increasing ray number N_R depends on the smoothness of the irradiation. For smooth irradiating intensity $\varepsilon_H \propto 1/N_R$, for piecewise sources $\varepsilon_H \propto 1/\sqrt{N_R}$ and for intensity fields field with derivative discontinuities $\varepsilon_H \propto 1/N_R$. The surface discretisation error is generally small compared with ε_H and can be reduced by refinement of the surface mesh. Our predictive experience is that when an adequately refined mesh and a sufficiently large number of rays are used the standard method produces results which are comparable in accuracy with Monte Carlo solutions. Our tests have shown that, for non-scattering, absorption/emission-only problems, the DTM is around 500 times faster than MC in terms of CPU times. For isotropic scattering, absorbing and emitting problems DTM is about 10 times faster than MC calculations (see Henson and Malalasekera, 1997a).

Further details of the basic method can be found in Shah (1979), Lockwood and Shah (1981) and Henson (1998). The discrete transfer method is mathematically simple and also applicable to any geometrical configuration. When a fast and efficient ray tracing algorithm is used the method is computationally efficient. It is well established in application to combustion-related problems, and extensions of the method to isotropic scattering problems and non-grey calculations have been demonstrated in Carvalho *et al*. (1991) and Henson and Malalasekera (1997b). The method is, however, not suitable for anisotropic scattering applications. Several enhancements and modifications to the original method have been proposed, e.g. Cumber (1995) and Coelho and Carvalho (1997).

13.4.3 Ray tracing

Both the MC and DTM require ray tracing in a given volume mesh. For Cartesian and cylindrical geometries the path of rays can be calculated using simple vector algebra. For non-orthogonal and unstructured mesh arrangements ray tracing requires some attention, particularly because the process has to be computationally efficient. For such applications a very efficient method is described in Malalasekera and James (1995, 1996), Henson and Malalasekera (1997a) and Henson (1998). This method has been successfully used to generate discrete transfer and MC solutions in complex geometries with non-orthogonal structured mesh systems. The method is also suitable for unstructured mesh arrangements, including those with mixed elements (tetrahedral and hexahedral elements).

13.4.4 The discrete ordinates method

In the discrete ordinates method the equation of transfer is solved for a set of n different directions in the total of 4π solid angle, and the integrals over directions are replaced by numerical quadrature. Thus the equation of transfer is approximated by

$$\frac{\mathrm{d}I(\mathbf{r},\ \mathbf{s}_i)}{\mathrm{d}s} = \kappa I_b(\mathbf{r}) - \beta I(\mathbf{r},\mathbf{s}_i) + \frac{\sigma_s}{4\pi}\sum_{j=1}^{n} w_j I_{-}(\mathbf{s}_j)\,\Phi(\mathbf{s}_i,\mathbf{s}_j)$$

$$i = 1, 2, \ldots, n \tag{13.35}$$

where w_j are quadrature weights associated with the directions $\mathbf{s}_j$ (see Modest, 2003). The equation is subject to the boundary condition

$$I(\mathbf{r}_w, \mathbf{s}_i) = \varepsilon(\mathbf{r}_w) I_b(\mathbf{r}_w) + \frac{1-\varepsilon(\mathbf{r}_w)}{\pi}\sum_{j=1}^{n} w_j I_{-}(\mathbf{r}_w, \mathbf{s}_j)\,|\mathbf{n}\,.\,\mathbf{s}_j| \tag{13.36}$$

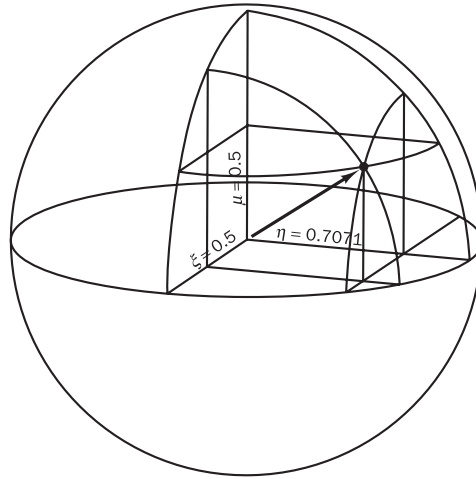
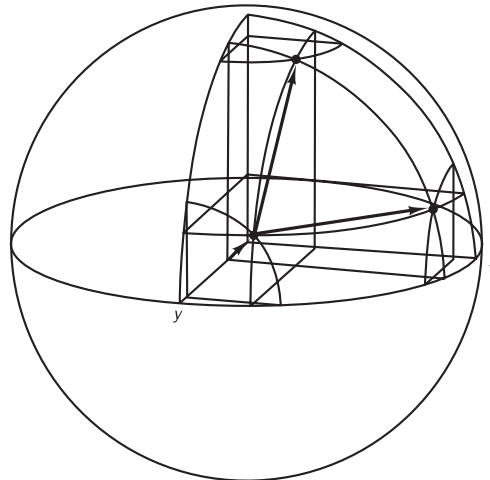
The angular ordinates $\mathbf{s}_j = \xi\mathbf{i} + \eta\mathbf{j} + \mu\mathbf{k}$ and angular weights w_j are available in Lathrop and Carlson (1965), Fiveland (1991), and the basis of obtaining the weights is discussed further in Modest (2003). The order of the S_N approximation is denoted by S_2, S_4, S_6 . . . S_N. The total number of directions used (n) is related to N through the relation $n = N(N + 2)$.

Direction cosines ξ, η or μ and weights w_j for basic discrete ordinates approximations are shown in Table 13.1. Only the positive direction cosines

Table 13.1 Ordinate directions and weights for S_2 and S_4 approximations

S_N *approximation*	*Ordinates*			*Weights*
	ξ	η	μ	
S_2 (symmetric)	0.5773	0.5773	0.5773	1.57079
S_2 (non-symmetric)	0.5000	0.7071	0.5000	1.57079
S_4	0.2959	0.2959	0.9082	0.52359
	0.2959	0.9082	0.2959	0.52359
	0.9082	0.2959	0.2959	0.52359

and weights are shown. The most basic discrete ordinate approximation is S_2. In S_2, two different direction cosines are used to define a principal direction. Therefore one direction is used in an eighth of a sphere. In total 2(2+2) or 8 directions are used per sphere. There are two S_2 representations: symmetric and non-symmetric. The symmetric representation uses equal values for all direction cosines whereas the non-symmetric representation uses different values. Figure 13.8a illustrates the non-symmetric S_2 representation in one-eighth of a sphere.

Figure 13.8 Discrete ordinates in one-eighth of a sphere: (a) illustration of the S_2 (non-symmetric) representation; (b) illustration of the S_4 representation

The next improved approximation is S_4. Here four different direction cosine values are used. They are ±0.2959 and ±0.9082 for ξ, μ or η. Using two different positive values we can generate three principal directions in one-eighth of a sphere as illustrated in Figure 13.8b. Only three directions are shown; all other directions may be obtained by using appropriate negative

and positive values, giving a total of 4(4+2) = 24 permutations. The angular discretisation represents the solid angle subtended by a sphere. Therefore, the sum of the weights for all directions should be equal to the solid angle of a sphere, i.e. 4π. This can be easily verified in Table 13.1 for the S_2 and S_4 approximations.

More detailed tabulations of ordinates including S_N approximations with $N > 4$ can be found in Siegel and Howell (2002) and Modest (2003).

The angular approximation transforms the original integro-differential equation into a set of coupled differential equations. For Cartesian co-ordinates equation (13.34) may be discretised as follows:

$$\xi_i \frac{\mathrm{d}I_i}{\mathrm{d}x} + \eta_i \frac{\mathrm{d}I_i}{\mathrm{d}y} + \mu_i \frac{\mathrm{d}I_i}{\mathrm{d}z} + \beta I_i = \beta S_i \quad i = 1, 2, \ldots, n \tag{13.37}$$

where ξ_i, η_i and μ_i are the direction cosines of direction i and

$$S_i = (1 - \omega)I_b + \frac{\omega}{4\pi}\sum_{j=1}^{n} w_j I_j \Phi_{ij} \quad i = 1, 2, \ldots, n \tag{13.38}$$

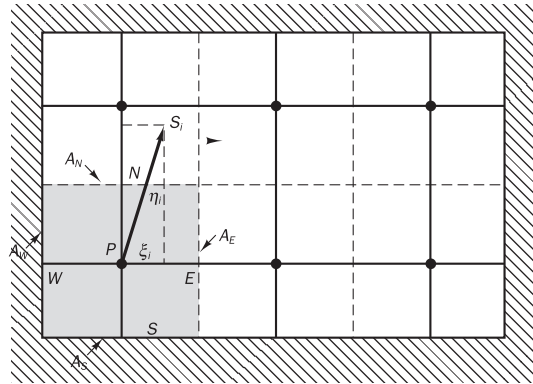
The set of coupled differential equations is solved by discretisation using the finite volume method: see Modest (2003). For example, consider the two-dimensional control volume shown in Figure 13.9. The components $\mathrm{d}I_i/\mathrm{d}x$, $\mathrm{d}I_i/\mathrm{d}y$, $\mathrm{d}I_i/\mathrm{d}z$ of the intensity gradient and other terms of equation (13.37) are integrated over the control volume, applying the usual finite volume approximations. Using the areas shown in Figure 13.9 we obtain

$$\begin{aligned}\xi_i(I_{E_i}A_E - I_{W_i}A_W) + \eta_i(I_{N_i}A_N - I_{S_i}A_S)\\ = -\beta I_{P_i}(\Delta V) + \beta S_{P_i}(\Delta V) \quad i = 1, 2, \ldots, n\end{aligned} \tag{13.39}$$

where I_{P_i} and S_{P_i} are volume averages of the intensity and source function. The intensity I_{P_i} at the centre of the cell is approximated as

$$I_{P_i} = \gamma I_{E_i} + (1 - \gamma)I_{W_i} \tag{13.40a}$$

$$I_{P_i} = \gamma I_{N_i} + (1 - \gamma)I_{S_i} \tag{13.40b}$$

Figure 13.9 A general two-dimensional geometry to illustrate the discrete ordinates method

The parameter γ is a weighting factor used to relate cell edge intensities to the volume average intensity. The weighting factor γ is a constant $0 \le \gamma \le 1$. The most widely used diamond and step difference schemes are obtained by setting γ to 0.5 and 1.0, respectively.

The solution marches through the computational domain as follows. At a boundary some I_i values are known from boundary conditions. Consider for example the two-dimensional corner cell shown in Figure 13.9. I_{W_i} and I_{S_i} are known for this cell. Using equations (13.40a) and (13.40b) we may write

$$\gamma I_{E_i} = I_{P_i} - (1-\gamma) I_{W_i} \tag{13. 41a}$$

$$\gamma I_{N_i} = I_{P_i} - (1-\gamma) I_{S_i} \tag{13.41b}$$

Substitution of equations (13.41a and b) into equation (13.39) and rearranging gives

$$I_{P_i} = \frac{\beta(\Delta V)\gamma S_{P_i} + \xi_i A_{WE} I_{W_i} + \eta_i A_{SN} I_{S_i}}{\beta(\Delta V)\gamma + \xi_i A_E + \eta_i A_N} \tag{13.42a}$$

$$\text{where } A_{WE} = \gamma A_W + (1-\gamma) A_E \tag{13.42b}$$

$$\text{and } A_{SN} = \gamma A_S + (1-\gamma) A_N \tag{13.42c}$$

Equation (13.42a) gives a means of calculating I_{P_i} from boundary intensities I_{W_i} and I_{S_i}. Once I_{P_i} is calculated, I_{E_i} and I_{N_i} can be obtained from equations (13.41a and 13.41b), and the process can move to the next cell where newly calculated values are used as boundary values for the next cell and so on. Starting from a boundary where the intensity is known the domain can be swept to find unknown intensities along each ordinate direction. The above procedure has to be repeated for all ordinate directions. For negative ordinate directions the process starts from the north and east boundaries. Iteration is required as initial boundary intensities are based only on approximate values (usually calculated using surface temperatures) and can only be updated once all incoming intensities are known.

Equation (13.42a) may be generalised to three dimensions as follows:

$$I_{P_i} = \frac{\beta(\Delta V)\gamma S_{P_i} + |\xi_i| A_x I_{x_i,i} + |\eta_i| A_y I_{y_i,i} + |\mu_i| A_z I_{z_i,i}}{\beta(\Delta V)\gamma + |\xi_i| A_x + |\eta_i| A_y + |\mu_i| A_z} \tag{13.43a}$$

where ΔV is the volume of the cell. Absolute values of direction cosines are used in the above equation to indicate that the equation is valid for both positive and negative direction cosines, and

$$A_x = (1-\gamma) A_{x_e} + \gamma A_{x_i} \tag{13.43b}$$

$$A_y = (1-\gamma) A_{y_e} + \gamma A_{y_i} \tag{13.43b}$$

$$A_z = (1-\gamma) A_{z_e} + \gamma A_{z_i} \tag{13.43c}$$

where subscripts i and e denote entering and exit faces of a control volume, respectively, and A_x, A_y and A_z refer to x, y and z control volume areas in a three-dimensional Cartesian control volume. As before, we start from boundary cells to calculate I_{Pi} and progress into inner cells. Once all the directional intensities are known, the radiative flux at a surface may be calculated from

$$q_-(\mathbf{r}) = \int_{2\pi} I_-(\mathbf{r}, \mathbf{s}) \mathbf{n} \,.\, \mathbf{s} d\Omega = \sum_{i=1}^{n} w_i I_i(\mathbf{r}, \mathbf{s}_i) \mathbf{n} \,.\, \mathbf{s}_i \tag{13.44}$$

The radiative source term for the enthalpy equation may be calculated from

$$S_{h,rad} = \nabla \cdot \mathbf{q}_r(\mathbf{r}) = 4\pi\kappa I_b - \kappa\sum_{i=1}^{n} w_i I_i(\mathbf{r}, \mathbf{s}) \tag{13.45}$$

Further details of the method and its derivation can be found in Modest (2003), Fiveland (1982, 1988, 1991), Fiveland and Jessee (1994), Jamaluddin and Smith (1988) and Hyde and Truelove (1977). The standard discrete ordinates method is suitable for Cartesian and axisymmetric geometries, but it is not directly applicable to non-orthogonal and unstructured grids. However, a modified version of the discrete ordinates method that is suitable for complex geometries has been demonstrated by Charette *et al*. (1997) and Sakami *et al*. (1996, 1998).

13.4.5 The finite volume method

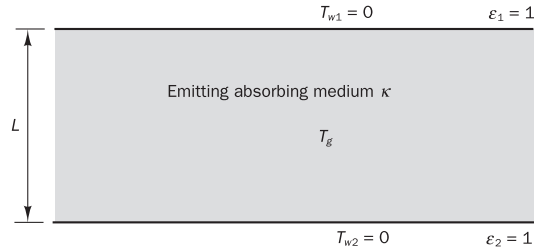
Raithby and co-workers (Raithby and Chui, 1990; Chui *et al*., 1992, 1993; Chui and Raithby, 1993) have presented a control volume integration method for the calculation of radiative heat transfer that shares several features with the discrete ordinates method. Equations for intensity are solved for a set of discrete directions, which span the total solid angle 4π. In addition to control volume integration the finite volume method uses a control angle integration as well (see Chai *et al*., 1994a, b). Further improvements to the original method along with appropriate interpolation techniques for non-orthogonal mesh systems are available for radiative heat transfer calculations in complex geometries. The difficulty that arises in the application of this class of methods to non-orthogonal cells is the evaluation of cell intensities and handling of control angle overhanging. These are both caused by misalignment of cell surfaces with discretised control angles. Several techniques have been developed to overcome this problem: details can be found in Chai and Modar (1996), Baek and Kim (1998), Baek *et al*. (1998) and Murthy and Mathur (1998). An illustrative program that demonstrate the finite volume method is available in the appendices of Modest (2003).

13.5 Illustrative examples

Example 13.1 One-dimensional emitting/absorbing problem with prescribed temperature

First we consider the one-dimensional radiative heating of two cold black plates by a layer of emitting/absorbing hot gas at temperature T_g and with absorption coefficient κ. The problem considered is shown in Figure 13.10. An analytical solution is available for this very simple problem: see e.g. Shah (1979), Siegel and Howell (2002) or Modest (2003). We apply the MC, DTM and discrete ordinates methods to calculate the heat flux to the plate surfaces and compare the solutions obtained with the analytical result. The non-dimensional wall heat flux is given by

$$\frac{q}{\sigma T_g^4} = 1 - 2E_3(\kappa L) \tag{13.46}$$

Figure 13.10 Configuration for one-dimensional temperature prescribed problem

where $E_3(\kappa L)$ is an exponential integral, values of which can be found in tabulated form in Siegel and Howell (2002), and L is the distance between the two plates. In this expression the wall heat flux has been non-dimensionalised by means of the black-body emissive power of the gaseous medium. We have computed solutions for optical path lengths $\kappa L = 0.1, 0.2, \ldots, 3.0$; this ranges from 'almost transparent (optically thin)' to 'optically dense (optically thick)'.

DTM and MC methods

Although the problem is essentially one-dimensional, the DTM and MC methods solve the directional intensity distribution by ray tracing, which must be applied in a three-dimensional fashion. Simple trigonometric expressions are sufficient to determine all ray intersections in this problem. For the DTM the solid angle was discretised using 10×10 for the N_θ and the N_ϕ directions; hence, 100 rays were used per surface in the calculation. For the MC solution the number of bundles fired was increased from 6×10^5 to 2×10^7 as the optical path length of the medium increased from $\kappa L = 0.1$ to 3.0. The standard error, S_n, was of the order of 10^{-4} for each MC simulation performed. Table 13.2 shows the level of error and CPU time used for MC solutions. CPU time depends on the computer system, but all DTM solutions used less than 1 CPU second, which highlights the cost of MC solutions.

Table 13.2 MC statistics

Optical length	*Total number of bundles*	*Standard error S_n*	*CPU time*
0.1	6.80×10^5	8.50×10^4	44
0.2	1.30×10^6	2.10×10^4	98
0.5	3.40×10^6	4.60×10^4	320
1.0	6.80×10^6	3.70×10^4	564
1.5	1.00×10^7	3.30×10^4	800
2.0	1.30×10^7	4.80×10^4	874
2.5	1.70×10^7	3.60×10^4	1047
3.0	2.00×10^7	3.30×10^4	1430

Discrete ordinates method (DOM)

The DOM solution to this simple problem can be obtained by applying equations (13.37) and (13.38). The solution procedure and the S_2 and S_4 discrete ordinates solutions are available in Modest (2003).

Using the S_2 approximation the radiative heat flux to walls in non-dimensional form is given by

$$\frac{q}{\sigma(T_w^4 - T_g^4)} = \frac{1}{\pi}\sum_{i=1}^{N/2} w_i' \mu_i \{\exp(-\tau/\mu_i) - \exp[-(\tau_L - \tau)/\mu_i]\} \quad (13.47)$$

where $\tau = \kappa y$ and $\tau_L = \kappa L$

Here κ is the absorption coefficient, y is the co-ordinate direction perpendicular to plates, and w_i' are appropriately summed weights for this one-dimensional problem. For example, for the non-symmetric S_2 approximation $\mu_1 = 0.5$, the associated weight is $\pi/2 = 1.5707$, and four μ-values are involved, giving the summed weights $w_i' = 4 \times \pi/2 = 2\pi$.

Using the non-symmetric S_2 approximation the non-dimensional heat flux is given by

$$\frac{q}{\sigma(T_w^4 - T_g^4)} = \{\exp(-\tau/0.5) - \exp[-(\tau_L - \tau)/0.5]\} \quad (13.48)$$

With the S_4 discrete ordinates approximation the non-dimensional heat flux is given by

$$\frac{q}{\sigma(T_w^4 - T_g^4)} = 0.3945012\{\exp(-\tau/0.2958759) - \exp[-(\tau_L - \tau)/0.2958759]$$

$$+ 0.6054088[\exp(-\tau/0.9082483) - \exp[-(\tau_L - \tau)/0.9082483]\} \quad (13.49)$$

The lower plate solution ($y = 0$) using the S_2 approximation is as follows:

$$\frac{q}{\sigma(T_w^4 - T_g^4)} = [1 - \exp(-\kappa L/0.5)] \quad (13.50)$$

and with S_4:

$$\frac{q}{\sigma(T_w^4 - T_g^4)} = 0.3945012[1 - \exp(-\kappa L/0.2958759)]$$

$$+ 0.6054088[1 - \exp(-\kappa L/0.9082483)] \quad (13.51)$$

Results

The predictions of the wall heat flux for all three solution methods are compared with the analytical solution in Table 13.3 and Figure 13.11. It can be seen that all solutions except the DOM S_2 appear to give good agreement. The MC solution is the most accurate of all. The S_2 discrete ordinates approximation is too simplistic to produce close agreement, but it does predict the correct trend. In terms of calculation cost the resources used by the DOM are negligible compared with the CPU times for the MC method.

Figure 13.11 Comparision of numerical results for one-dimensional temperature prescribed problem

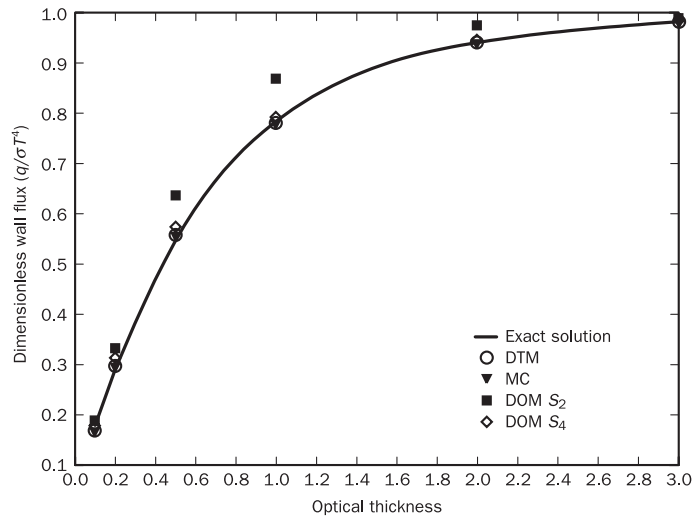



Table 13.3 Comparison of results from different solution methods for the one-dimensional temperature prescribed problem

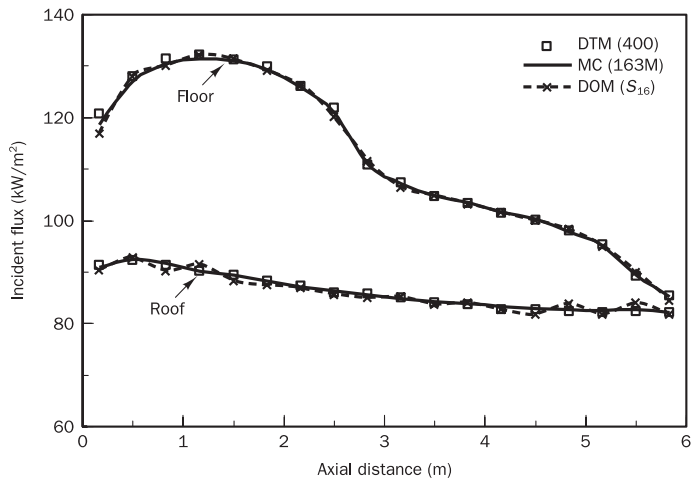
Optical thickness	*Exact solution*	*MC*	*DTM*	*DOM* S_2	*DOM* S_4
0.1	0.16742	0.167702	0.16923	0.18127	0.17626
0.2	0.29779	0.296686	0.29610	0.32968	0.31349
0.5	0.55732	0.557363	0.55680	0.63212	0.57800
1.0	0.78076	0.780648	0.78062	0.86467	0.78516
1.5	0.88551	0.886071	0.88653	0.95021	0.88134
2.0	0.93970	0.939207	0.93974	0.98168	0.93251
2.5	0.96737	0.966947	0.96740	0.99326	0.96122
3.0	0.98211	0.981435	0.98214	0.99752	0.97763

Example 13.2 Radiation in a three-dimensional furnace geometry

Next we consider the much more challenging, but realistic problem of the three-dimensional IFRF furnace geometry, which has been widely used in the radiative heat transfer literature to evaluate calculation methods. The configuration is outlined in detail in the paper by Jamaluddin and Smith (1988). In this illustration we apply the MC, DTM and S_{16} DOM methods for the calculation of radiative flux to the floor and roof of the furnace, described in the IFRF-M3 Trial (Flame 10). The furnace dimensions are 6.0 m × 2.0 m × 2.0 m in the x-, y- and z-directions and the measured temperatures by Hyde and Truelove (1977) are used to define the radiation problem. In the absence of an analytical solution, the MC results along with the standard error estimate are used as the benchmark solution. The discrete transfer calculation has been performed with 400 rays per surface element, while S_{16} quadrature is used for the discrete ordinates solution.

Solution

Figure 13.12 shows the calculated incident radiative flux on the floor and the roof of the furnace, and it is clear that the differences between the 400-ray discrete transfer, the S_{16} discrete ordinates and the MC solutions are very small. A closer look at the actual numerical values (not presented here) reveals that the discrete ordinate solution has the highest deviation. This could be attributed to its spatial differencing practice and the accuracy of the quadrature set. Malalasekera *et al*. (2002) have also shown that lower-order discrete ordinates solutions do not provide good agreement with MC results. This example illustrates that careful selection of numerical discretisation parameters for the DTM and DOM can yield solutions that are very close to MC solutions. It is worth noting that MC solutions require a vast amount of computer resources (more than 20 times larger compared with DTM and DOM) and therefore are not suitable for practical calculations involving CFD.

Figure 13.12 Calculated incident radiative flux on the floor and the roof of the furnace. DTM (400) is the discrete transfer method solution with 400 rays per point, MC (163M) is the Monte Carlo solution with 163 million bundles, and DOM (S16) is the discrete ordinates S_{16} solution

Example 13.3 Radiation in three-dimensional complex geometry

In this example we present a further comparison of methods in the L-shaped geometry shown in Figure 13.13. This more complex geometry has also been used in the literature to evaluate different radiation calculation techniques (Malalasekera and James, 1995, 1996; Henson and Malalasekera, 1997a; Sakami *et al*, 1998; Hsu and Tan, 1997). The problem specification is: (i) all walls are black at a temperature of 500 K and (ii) the enclosure is filled with an emitting/absorbing and non-scattering medium at a temperature of 1000 K. The L-shape presents additional complexity for solution methods because in a CFD computation the geometry cannot be discretised using Cartesian co-ordinates without unnecessarily wasting storage by blocking off a large number of cells. For the application of the DTM the geometry has been modelled using the non-orthogonal surface mesh shown in Figure 13.13. The standard discrete ordinates method is not applicable when a non-orthogonal mesh is employed. Here we have used a special version of the discrete ordinates method by Sakami *et al* (1998). Figure 13.14 shows the predicted distribution of net radiative heat flux along line B–B (Figure 13.13) for different values of the absorption coefficient $\kappa = 0.5$, 1.0, 2.0, 5.0 and 10.0 using the MC, DTM (16×16 rays per hemisphere), DOM method (with the

Figure 13.13 L-shaped geometry

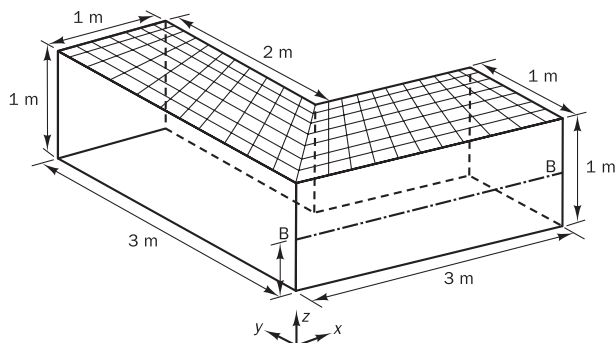
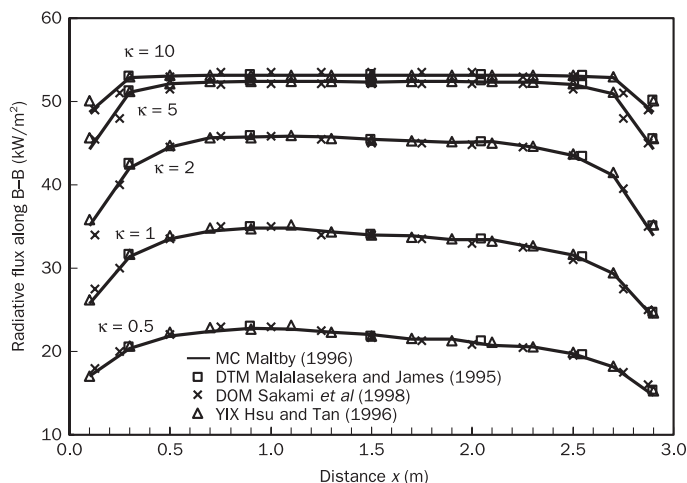



Figure 13.14 Comparison of numerical results for the L-shaped geometry

S_4 approximation), and also a general-purpose integral equation technique known as the YIX method (Hsu *et al*, 1993). The discrete transfer solution is from Malalasekera and James (1995), the MC results are by Maltby (1996) and the YIX solution by Hsu and Tan (1997). The MC calculation has been performed using a mesh of 204 volumes using symmetry and releasing 1 024 000 bundles per volume or per surface. It can be seen that results of all methods agree very well. The example is a good illustration to show the flexibility of the DTM and MC methods, which can both be applied without any modifications to complex meshes. The DOM requires a special formulation. When adequate resources are used all methods produce equally good results that are in agreement with the MC solution. Further applications of the DTM and MC method in complex geometry applications can be found in Henson (1998), Henson and Malalasekera (1997a), Malalasekera and James (1995, 1996) and Hsu and Tan (1997).

13.6 Calculation of radiative properties in gaseous mixtures

In many practical situations where radiative heat transfer is important, e.g. furnaces and combustors, the participating medium consists of a gaseous mixture of combustion products. Its radiative properties, such as the absorption coefficient, required for radiation calculations have to be evaluated by

considering the composition, temperature and pressure of the mixture. Accurate calculations have to take into account the spectral characteristics of radiative properties but, for practical purposes, total emissivity values and scattering coefficients can be used. There are a number of well-established techniques of estimating total properties, including spectral band models, correlations for gas absorptivities and emissivities, *k*-distribution methods and weighted-sum-of-grey-gases model. The most widely used approximate radiative property calculation method for combustion problems is the weighted-sum-of-grey-gases (WSGG) model (see above references). Another popular model in the radiation literature is the RADCAL program of Groshandler (1993), which takes into account detailed spectral properties. Detailed discussions of the techniques can be found in Siegel and Howell (2002), Brewster (1992), Denison and Webb (1993, 1995) and Modest (2003).

When radiation is involved in combustion simulations the overall success of predictions depends on many modelling aspects: flow and the turbulence model, the combustion model, degree of detailed chemistry included, the radiative property calculation methods and the radiation model. All these are coupled and, therefore, given the available resources, the best possible submodelling practice should be used to achieve a good overall result. More advanced property calculations and finer radiation calculations will give the benefit of higher accuracy and better resolution in radiation source terms. The level of detail required in the application and affordable cost of a calculation is the major factor which determines the choice. It is well known that temperatures are overpredicted if radiation is neglected. As minor species such as NO are very sensitive to temperature, CFD combustion predictions need to include radiation in order to achieve accurate predictions of these pollutant species. In order to save computing resources it is sometimes possible to carry out the radiation calculations on a much coarser mesh than the one used for the fluid flow and combustion calculations. The source terms for the fluid flow calculation can be appropriately interpolated from the coarse mesh calculation. This practice is quite adequate in many situations, saving considerable computational overheads.

13.7 Summary

Most commercial CFD packages provide the user with the option of one or more of the radiation models described in this chapter. Their main features can be summarised as follows:

- *Monte Carlo method*: the most general and versatile of all models. It requires very large amounts of computing resources and is, therefore, unsuitable for general-purpose CFD calculations. The probabilistic basis of the method enables its users to generate error estimates. The MC method is useful in situations where solutions with quantifiable uncertainty are required, e.g. for benchmarking and validation of other models.
- *Discrete transfer method*: an economical general-purpose algorithm based on ray tracing. The DTM has been successfully applied to a wide range of combusting flows. It is suitable for all types of structured and unstructured meshes. The DTM is limited to isotropic scattering, and is non-conservative (non-zero sum of the radiative heat fluxes incident upon the bounding surfaces of an enclosure); Coelho and Carvalho

(1997) have proposed a conservative DTM formulation. Uncertainty estimates are available for a limited number of simple cases.

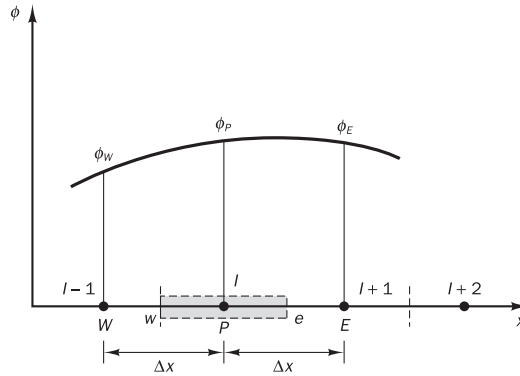
- *Discrete ordinates method*: a method based on the numerical quadrature of angular direction integration and control volume integration for path integration. The original method was limited to Cartesian and axisymmetric grids, but the Sakami *et al*. (1996, 1998) DOM enables applications involving unstructured meshes and more complex geometry. The method in general is efficient and applicable to scattering and non-scattering problems. Higher-order discrete ordinates are required to achieve accuracy and one of the drawbacks of the method is that it is non-conservative and can suffer from what are known as 'ray effects' mainly due to the discretisation practice used in the formulation.
- *Finite volume method*: a method based on integration over the control angle for direction integration and the control volume method for path integration developed by Raithby and co-workers. The main advantage of this method is that it is fully conservative and applicable to arbitrary geometries. One of the disadvantages is that the method can incur errors due to what is known as 'solid angle overhanging'.

In this chapter we have presented the key aspects of the more popular methods, which should clarify the role played by the numerical parameters associated with each model (ray number, order of discrete ordinates, grey gas models etc.). The interested reader should follow up the relevant literature to understand the finer details of these methods and also to research the many other methods that we were forced to omit for the sake of brevity.

Appendix A Accuracy of a flow simulation

In the derivations of Chapter 4 a linear profile assumption was used to calculate the gradients $(\partial\phi/\partial x)$, $(\partial\phi/\partial y)$ etc. at the faces of the control volume. For simple diffusion problems the practice was shown to give reasonably accurate results even for coarse grids. In Example 4.3 we also observed that by refining the grid the accuracy of the solution can be improved. Grid refinement is the main tool at the disposal of the CFD user for the improvement of the accuracy of a simulation. The user would typically perform a simulation on a coarse mesh first to get an impression of the overall features of the solution. Subsequently the grid is refined in stages until no (significant) differences of results occur between successive grid refinement stages. Results are then called 'grid independent'. Here we briefly demonstrate the theoretical basis of this method of accuracy improvement and compare the *order* of discretisation schemes as a measure their efficacy.

Consider the equally spaced one-dimensional grid (spacing Δx) shown in Figure A.1.

Figure A.1

For a function $\phi(x)$ the Taylor series development of $\phi(x + \Delta x)$ around the point i at x is

$$\phi(x + \Delta x) = \phi(x) + \left(\frac{\partial\phi}{\partial x}\right)_x \Delta x + \left(\frac{\partial^2\phi}{\partial x^2}\right)_x \frac{\Delta x^2}{2} + \ldots \tag{A.1}$$

In our notation we use discrete values ϕ_P and ϕ_E for $\phi(x)$ and $\phi(x + \Delta x)$ respectively so that equation (A.1) can be written as

$$\phi_E = \phi_P + \left(\frac{\partial\phi}{\partial x}\right)_P \Delta x + \left(\frac{\partial^2\phi}{\partial x^2}\right)_P \frac{\Delta x^2}{2} + \ldots \tag{A.2}$$

This may be rearranged to give

$$\left(\frac{\partial\phi}{\partial x}\right)_P \Delta x = \phi_E - \phi_P - \left(\frac{\partial^2\phi}{\partial x^2}\right)_P \frac{\Delta x^2}{2} - \ldots \tag{A.3}$$

$$\left(\frac{\partial\phi}{\partial x}\right)_P = \frac{\phi_E - \phi_P}{\Delta x} - \left(\frac{\partial^2\phi}{\partial x^2}\right)_P \frac{\Delta x}{2} - \ldots$$

So

$$\left(\frac{\partial\phi}{\partial x}\right)_P = \frac{\phi_E - \phi_P}{\Delta x} + \text{Truncated terms} \tag{A.4}$$

By neglecting the truncated terms which involve the multiplying factor Δx we may write

$$\left(\frac{\partial\phi}{\partial x}\right)_P \approx \frac{\phi_E - \phi_P}{\Delta x} \tag{A.5}$$

The error involved in the approximation (A.5) is due to neglect of the truncated terms. Formula (A.3) suggests that the truncation error can be reduced by decreasing Δx. In general the truncated terms of a finite difference scheme contain factors Δx^n. The power n of Δx governs the rate at which the error tends to zero as the grid is refined and is called the **order of the difference approximation**. Therefore equation (A.5) is said to be first-order in Δx and we write

$$\boxed{\left(\frac{\partial\phi}{\partial x}\right)_P = \frac{\phi_E - \phi_P}{\Delta x} + O(\Delta x)} \tag{A.6}$$

Since it uses values at points E and P (where $x_E > x_P$) to evaluate the gradient $(\partial\phi/\partial x)$ at P, formula (A.6) is called a forward difference formula with respect to point P.

Similarly we may derive a backward difference formula for $(\partial\phi/\partial x)$ at P from

$$\phi(x - \Delta x) = \phi(x) - \left(\frac{\partial\phi}{\partial x}\right)_x \Delta x + \left(\frac{\partial^2\phi}{\partial x^2}\right)_x \frac{\Delta x^2}{2} + \ldots \tag{A.7}$$

After some algebra we find the backward difference formula for $(\partial\phi/\partial x)$ at P:

$$\boxed{\left(\frac{\partial\phi}{\partial x}\right)_P = \frac{\phi_P - \phi_W}{\Delta x} + O(\Delta x)} \tag{A.8}$$

Equations (A.7) and (A.8) are both first-order accurate. The backward and forward difference formulae described here involve values of ϕ at two points only.

By subtracting equation (A.7) from (A.1) we get

$$\phi(x + \Delta x) - \phi(x - \Delta x) = 2\left(\frac{\partial\phi}{\partial x}\right)_P \Delta x + \left(\frac{\partial^3\phi}{\partial x^3}\right)_P \frac{\Delta x^3}{3!} + \ldots \tag{A.9}$$

A third formula for $(\partial\phi/\partial x)_P$ can be obtained by rearranging equation (A.9) as

$$\left(\frac{\partial\phi}{\partial x}\right)_P = \frac{\phi_E - \phi_W}{2\Delta x} + O(\Delta x^2) \tag{A.10}$$

Equation (A.10) uses values at E and W to evaluate the gradient at the midpoint P, and is called a central difference formula. The central differencing formula is second-order accurate. The quadratic dependence of the error on grid spacing means that after grid refinement the error reduces more quickly in a second-order accurate differencing scheme than in a first-order accurate scheme.

In the finite volume discretisation procedure developed in section 4.2 the gradient at a cell face, e.g. at 'e', was evaluated using

$$\left(\frac{\partial\phi}{\partial x}\right)_e = \frac{\phi_E - \phi_P}{\Delta x} = \frac{\phi_E - \phi_P}{2(\Delta x/2)} \tag{A.11}$$

By comparing formulae (A.10) and (A.11) it can be easily recognised that (A.11) evaluates the gradient at the midpoint between P and E through a central difference formula at point 'e': therefore its accuracy is second-order for uniform grids.

It is relatively straightforward to demonstrate the third-order accuracy of the QUICK differencing scheme for the convective flux at a midpoint cell face in a uniform grid of spacing Δx. The QUICK scheme calculates the value ϕ_e at the east cell face of a general node as

$$\phi_e = \frac{3}{8}\phi_E + \frac{6}{8}\phi_P - \frac{1}{8}\phi_W \tag{A.12}$$

Taylor series expansion about the east face gives

$$\phi_E = \phi_e + \frac{1}{2}\Delta x\left(\frac{\partial\phi}{\partial x}\right)_e + \frac{1}{2}\left(\tfrac{1}{2}\Delta x\right)^2\left(\frac{\partial^2\phi}{\partial x^2}\right)_e + O(\Delta x^3) \tag{A.13}$$

$$\phi_P = \phi_e - \frac{1}{2}\Delta x\left(\frac{\partial\phi}{\partial x}\right)_e + \frac{1}{2}\left(-\tfrac{1}{2}\Delta x\right)^2\left(\frac{\partial^2\phi}{\partial x^2}\right)_e + O(\Delta x^3) \tag{A.14}$$

$$\phi_W = \phi_e - \frac{3}{2}\Delta x\left(\frac{\partial\phi}{\partial x}\right)_e + \frac{1}{2}\left(-\tfrac{3}{2}\Delta x\right)^2\left(\frac{\partial^2\phi}{\partial x^2}\right)_e + O(\Delta x^3) \tag{A.15}$$

If we add together $3/8 \times$ (A.13) + $6/8 \times$ (A.14) − $1/8 \times$ (A.15) we obtain

$$\frac{3}{8}\phi_E + \frac{6}{8}\phi_P - \frac{1}{8}\phi_W = \phi_e + O(\Delta x^3) \tag{A.16}$$

The terms involving orders Δx and Δx^2 cancel out in this uniform grid and the QUICK scheme is a third-order accurate approximation.

If necessary formulae involving more points (in backward or forward directions) can be derived which have higher-order accuracy: see texts such as Abbott and Basco (1989) or Fletcher (1991) for further details.

Appendix B Non-uniform grids

For the sake of simplicity the worked examples have focused on uniform grids of nodal points. However, the derivation of discretised equations in Chapters 4 and 5 used general geometrical dimensions such as δx_{PE}, δx_{WP} etc. and is also valid for non-uniform grids. In a non-uniform grid the faces e and w of a general node may not be at the midpoints between nodes E and P, and nodes W and P, respectively. In this case the interface values of diffusion coefficients Γ are calculated as follows:

$$\Gamma_w = (1 - f_W)\Gamma_W + f_W\Gamma_P \tag{B.1a}$$

where the interpolation factor f_w is given by

$$f_W = \frac{\delta x_{Ww}}{\delta x_{Ww} + \delta x_{wP}} \tag{B.1b}$$

and

$$\Gamma_e = (1 - f_P)\Gamma_P + f_P\Gamma_E \tag{B.1c}$$

where

$$f_P = \frac{\delta x_{Pe}}{\delta x_{Pe} + \delta x_{eE}} \tag{B.1d}$$

For a uniform grid these expressions simplify to equations (4.5a–b): since $f_W = 0.5$, $f_P = 0.5$, we have $\Gamma_w = (\Gamma_W + \Gamma_P)/2$ and $\Gamma_e = (\Gamma_P + \Gamma_E)/2$.

Basically there are two practices used to locate control volume faces in non-uniform grids (Patankar, 1980):

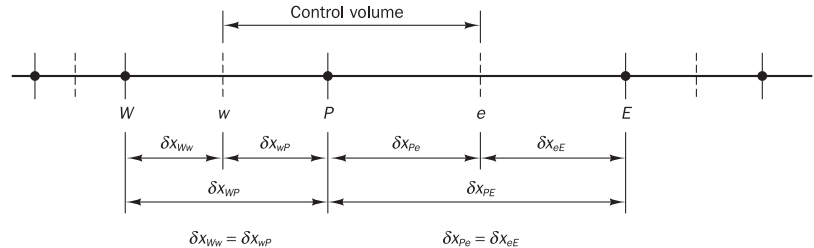
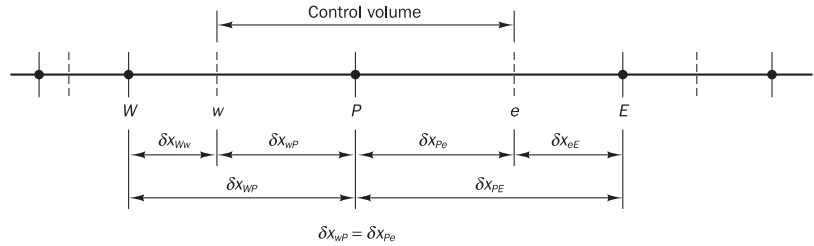
Practice A: Nodal points are defined first and the control volume faces located mid-way between the grid points. This is illustrated in Figure B.1.

Practice B: Locations of the control volume faces are defined first and the nodal points are placed at the centres of the control volumes. This is illustrated in Figure B.2.

Figure B.2

Here the faces of a control volume are not at the midpoint between the nodes. The evaluation of gradients obtained through a linear approximation is unaffected because the gradient remains the same at any point between the nodes in question, but the values of diffusion coefficient Γ need to be evaluated using interpolation functions (B.1).

It is very important to note that central difference formulae for the calculation of gradients at cell faces and the QUICK scheme for convective fluxes are only second- and third-order accurate respectively when the control volume face is mid-way between nodes. In practice A a control volume face, e for example, is mid-way between nodes P and E, so the differencing formula used to evaluate the gradient $(\partial\phi/\partial x)_e$ is second-order accurate. A further advantage of practice A is that property values Γ_e, Γ_w etc. can be easily evaluated by taking the average values. The disadvantage of practice A is that the value of the variable ϕ at P may not necessarily be the most representative value for the entire control volume as point P is not at the centre of the control volume. In practice B the value of ϕ at P is a good representative value for the control volume as P lies at the centre of the control volume, but the discretisation schemes lose accuracy. A thorough discussion of these two practices can be found in Patankar (1980), to which the reader is referred for further details.

Appendix C Calculation of source terms

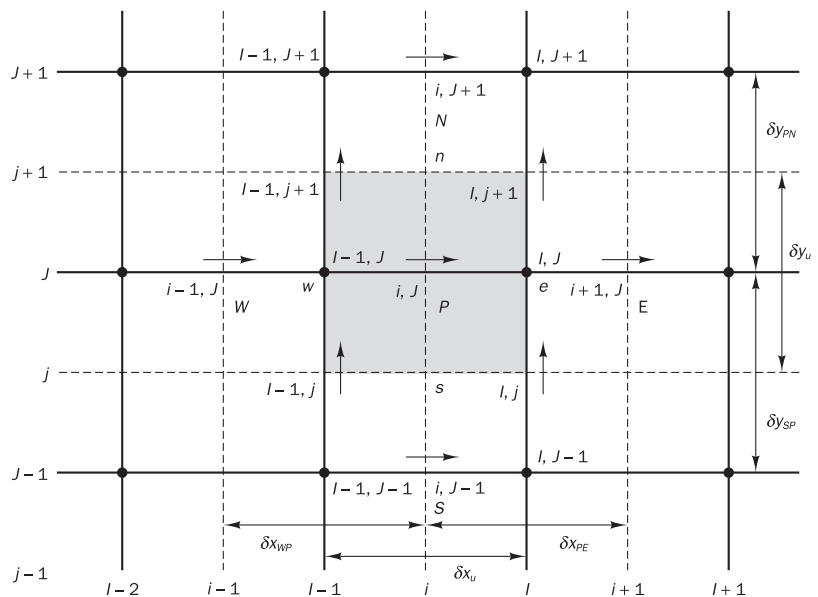
Source terms of the discretised equations are evaluated using prevailing values of the variables. Since a staggered grid is employed in the calculation procedures, interpolation is required in the calculation of velocity gradient terms which often appear in source terms. For example, consider the two-dimensional u-momentum equation

$$\frac{\partial}{\partial x}(\rho uu) + \frac{\partial}{\partial y}(\rho vu) = \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial u}{\partial y}\right) - \frac{\partial p}{\partial x} + S_u \quad \text{(C.1)}$$

where

$$S_u = \frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial x}\right)$$

A small part of the solution grid is shown in Figure C.1; we use the standard notation for the backward staggered velocity components introduced in Chapter 6.

Figure C.1

The discretised form of equation (C.1) for the u-control volume centred at (i, J) is

$$a_{i,J}u_{i,J} = \sum a_{nb}u_{nb} - \frac{(P_{I,J} - P_{I-1,J})}{\delta x_u}\Delta V_u + \bar{S}_u \Delta V_u \tag{C.2}$$

where δx_u is the width of the u-control volume and ΔV_u is the volume of the u-control volume.

The source term for the u-velocity equation at the cell shown is evaluated as

$$\begin{aligned}
\bar{S}_u \Delta V &= \left[\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left(\mu\frac{\partial v}{\partial x}\right)\right]_{Cell} \Delta V \\
&= \left[\frac{\left(\mu\frac{\partial u}{\partial x}\right)_e - \left(\mu\frac{\partial u}{\partial x}\right)_w}{\delta x_u} + \frac{\left(\mu\frac{\partial v}{\partial x}\right)_n - \left(\mu\frac{\partial v}{\partial x}\right)_s}{\delta y_u}\right]\Delta V \\
&= \left[\frac{\left(\mu\frac{u_{i+1,J} - u_{i,J}}{\partial x_{PE}}\right) - \left(\mu\frac{u_{i,J} - u_{i-1,J}}{\partial x_{WP}}\right)}{\delta x_u}\right. \\
&\quad \left. + \frac{\left(\mu\frac{v_{I,j+1} - v_{I-1,j+1}}{\partial x_u}\right) - \left(\mu\frac{v_{I,j} - v_{I-1,j}}{\partial x_u}\right)}{\delta y_u}\right]\delta x_u \delta y_u
\end{aligned} \tag{C.3}$$

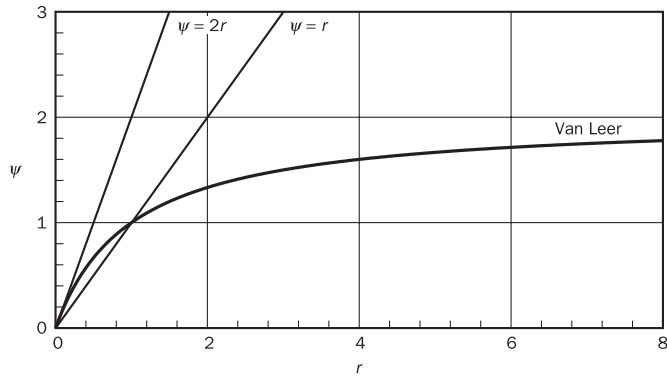
Source terms of other equations are calculated in a similar manner.

Appendix D Limiter functions used in Chapter 5[1]

1 Van Leer limiter (Van Leer, 1974):

$$\psi(r) = \frac{r + |r|}{1 + r} \tag{D.1}$$

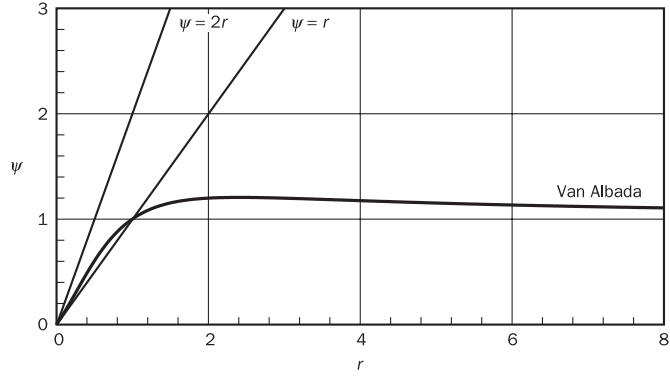
Figure D.1 shows this limiter on a r–ψ diagram. The limiter function stays in the TVD region and goes through the point (1, 1), so it is therefore a second-order accurate TVD limiter. The limiter tends to $\psi = 2$ for large values of r and is symmetrical.

Figure D.1 Van Leer limiter

2 Van Albada limiter (Van Albada *et al*, 1982):

$$\psi(r) = \frac{r + r^2}{1 + r^2} \tag{D.2}$$

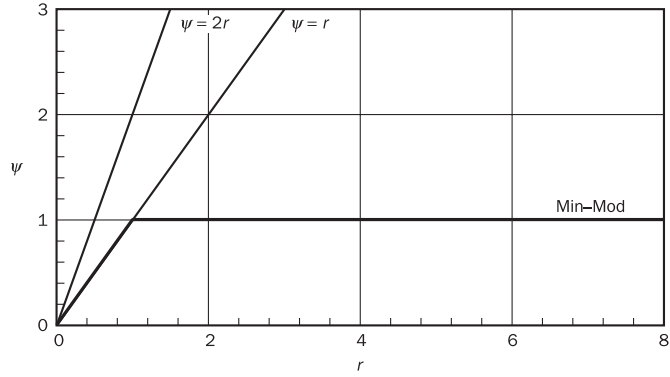
This function is shown in Figure D.2. This limiter also goes through (1, 1) and stays well below the upper limit for TVD schemes. As can be seen from Figure D.2, this limiter tends to $\psi = 1$ as $r \to \infty$. It is straightforward to check that the limiter is symmetrical.

[1] For details of references see Chapter 5.

Figure D.2 Van Albada limiter

3 Min–Mod limiter:

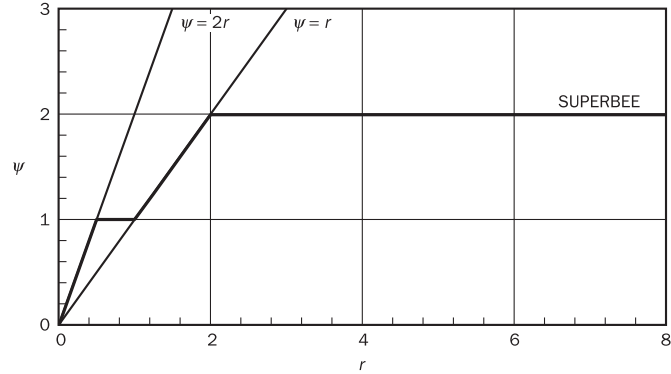
$$\psi(r) = \begin{cases} \min(r, 1) & \text{if } r > 0 \\ 0 & \text{if } r \leq 0 \end{cases} \tag{D.3}$$

Figure D.3 shows that this limiter is piecewise linear and follows the lowest boundary for a second-order accurate TVD scheme on a r–ψ diagram. The limiter is symmetrical.

Figure D.3 Min–Mod limiter

4 SUPERBEE limiter of Roe (1983):

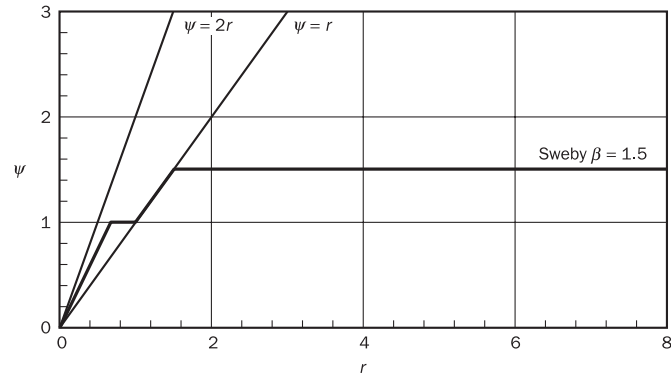
$$\psi(r) = \max[0, \min(2r, 1), \min(r, 2)] \tag{D.4}$$

Figure D.4 shows that this limiter is also a combination of linear expressions. The limiter is symmetrical and follows the upper limit of the TVD region on the r–ψ diagram.

Figure D.4 SUPERBEE limiter

5 Sweby limiter (Sweby, 1984): This limiter represents a generalisation of the Min–Mod and SUPERBEE limiters by means of a single parameter β. The limiter function is as follows:

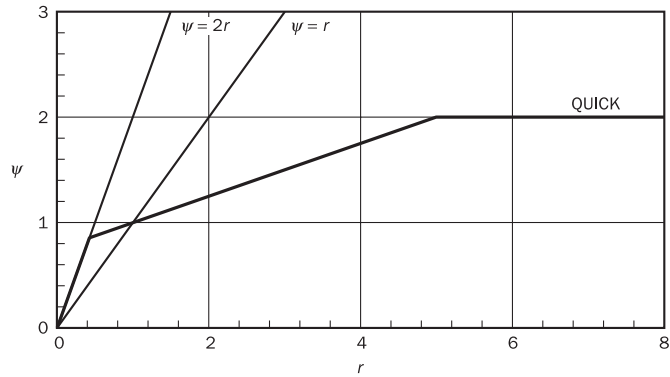
$$\psi(r) = \max[0, \min(\beta r, 1), \min(r, \beta)] \quad \text{(D.5)}$$

The limiter is symmetrical. We consider only the range of values $1 \leq \beta \leq 2$, and note that Sweby's limiter becomes the Min–Mod limiter when $\beta = 1$ and the SUPERBEE limiter of Roe when $\beta = 2$. The limiters span the whole TVD region between its upper and lower limits. Figure D.5 shows the $r-\psi$ diagram for Sweby's limiter when $\beta = 1.5$.

Figure D.5 Sweby limiter

6 QUICK limiter of Leonard (Leonard, 1988): A monotonic TVD version of the QUICK scheme can be achieved by the following function (see Lien and Leschziner, 1993):

$$\psi(r) = \max[0, \min(2r, (3 + r)/4, 2)] \quad \text{(D.6)}$$

Figure D.6 shows the r–ψ diagram for this limiter, which is **not symmetric**.

Figure D.6 QUICK limiter

7 UMIST limiter (Lien and Leschziner, 1993): UMIST, which stands for Upstream Monotonic Interpolation for Scalar Transport, is a symmetrical form of the QUICK limiter. The function is given by

$$\psi(r) = \max[0, \min(2r, (1 + 3r)/4, (3 + r)/4, 2)] \tag{D.7}$$

Figure D.7 shows this limiter on the r–ψ diagram.

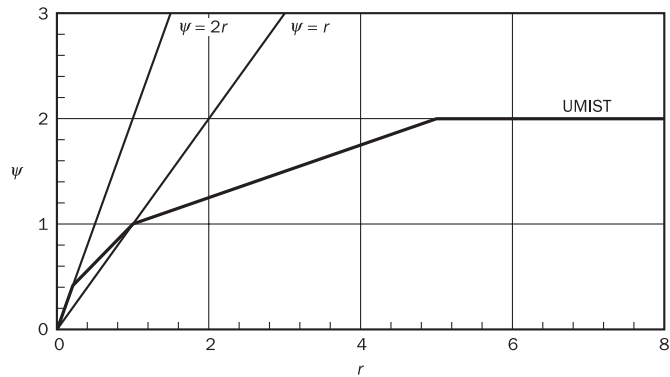
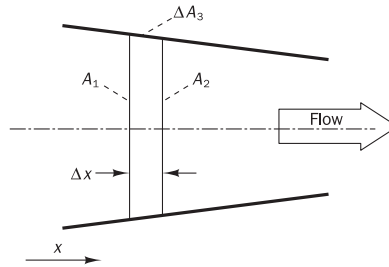
Figure D.7 UMIST limiter

Appendix E Derivation of one-dimensional governing equations for steady, incompressible flow through a planar nozzle

Figure E.1

We develop one-dimensional equations for frictionless, incompressible flow by application of the mass conservation and x-momentum equations from Chapter 2 to a small control volume ΔV of length Δx in the flow direction.

Mass conservation:

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{V}) = 0 \tag{2.4}$$

Since the flow is steady we have: $\operatorname{div}(\rho \mathbf{V}) = 0$ (E.1)

We integrate over the control volume and apply Gauss' divergence theorem (2.41) to convert the resulting volume integral into an integral over the bounding surfaces of the control volume:

$$\int_{\Delta V} \operatorname{div}(\rho \mathbf{V}) \mathrm{d}V = \int_{A_1} \rho \mathbf{V} \,.\, \mathbf{n} \mathrm{d}A + \int_{A_2} \rho \mathbf{V} \,.\, \mathbf{n} \mathrm{d}A + \int_{\Delta A_3} \rho \mathbf{V} \,.\, \mathbf{n} \mathrm{d}A = 0 \tag{E.2}$$

There is no mass flow across the side walls, so the last integral over ΔA_3 is equal to zero.

The velocity distribution is uniform and the velocity only changes as a function of the streamwise x-co-ordinate. The velocity vector $\mathbf{V}_1$ is in the positive x-direction, so $\mathbf{V}_1 = +u_1\mathbf{i}$, where u_1 is the magnitude of the x-velocity at A_1; similarly $\mathbf{V}_2 = +u_2\mathbf{i}$, where u_2 is the magnitude of the x-velocity at A_2. The outward normal vector $\mathbf{n}_1$ is in the negative x-direction, so $\mathbf{n}_1 = -\mathbf{i}$. The outward normal vector $\mathbf{n}_2$ on the other hand is in the positive x-direction, so $\mathbf{n}_2 = +\mathbf{i}$. Thus,

$$\int_{\Delta V} \text{div}(\rho \mathbf{V}) dV = -\rho u_1 A_1 + \rho u_2 A_2 = 0 \tag{E.3}$$

To develop the one-dimensional mass conservation equation we consider a control volume with sufficiently small length Δx so that we can use a one-term Taylor-series expansion to write expressions for velocity and cross-sectional area:

$$u_1 = u \quad \text{and} \quad u_2 = u + \frac{\mathrm{d}u}{\mathrm{d}x}\Delta x \tag{E.4a}$$

$$A_1 = A \quad \text{and} \quad A_2 = A + \frac{\mathrm{d}A}{\mathrm{d}x}\Delta x \tag{E.4b}$$

Substitution of equations (E.4a) and (E.4b) into (E.3) yields

$$-\rho u A + \rho\left(u + \frac{du}{dx}\Delta x\right)\left(A + \frac{dA}{dx}\Delta x\right) = 0 \tag{E.5}$$

Multiplying out the brackets and neglecting terms of order Δx^2, we obtain

$$\rho u \frac{\mathrm{d}A}{\mathrm{d}x}\Delta x + \rho A \frac{\mathrm{d}u}{\mathrm{d}x}\Delta x = 0 \tag{E.6}$$

Dividing the equation by the control volume length Δx and combining the two terms on the left-hand side yields

$$\boxed{\frac{\mathrm{d}(\rho u A)}{\mathrm{d}x} = 0} \tag{E.7}$$

The results (E.3) and (E.7) are familiar from introductory fluid mechanics texts; the former is the usual continuity equation for a macroscopic control volume and the latter is the differential form.

x-momentum conservation:

$$\frac{\partial(\rho u)}{\partial t} + \text{div}(\rho u \mathbf{V}) = -\frac{\partial p}{\partial x} + \text{div}(\mu \text{ grad}(u)) + S_u \tag{2.37a}$$

For steady, incompressible, frictionless flow this reduces to:

$$\text{div}(\rho u \mathbf{V}) = -\frac{\partial p}{\partial x} \tag{E.8}$$

Once more we integrate both sides of the equation over the control volume and apply Gauss's theorem to obtain integrals over the bounding surfaces. For the left-hand side we find:

$$\int_{\Delta V} \text{div}(\rho u \mathbf{V})\mathrm{d}V = \int_{A_1} \rho u \mathbf{V} \, . \, \mathbf{n}\mathrm{d}A + \int_{A_2} \rho u \mathbf{V} \, . \, \mathbf{n}\mathrm{d}A + \int_{\Delta A_3} \rho u \mathbf{V} \, . \, \mathbf{n}\mathrm{d}A$$

$$= -\rho u^2 A + \rho\left(u + \frac{\mathrm{d}u}{\mathrm{d}x}\Delta x\right)^2\left(A + \frac{\mathrm{d}A}{\mathrm{d}x}\Delta x\right) \tag{E.9}$$

Using (E.5) this can be rewritten as follows:

$$-\rho u^2 A + \rho\left(u + \frac{du}{dx}\Delta x\right)^2\left(A + \frac{dA}{dx}\Delta x\right)$$

$$= \rho u A\left(-u + u + \frac{du}{dx}\Delta x\right) = \rho u A\frac{du}{\partial x}\Delta x \qquad \text{(E.10)}$$

The control volume integral of the right-hand side of equation (E.8) and subsequent application of Gauss's divergence theorem (2.41) yields:

$$-\int_{\Delta V}\frac{\partial p}{\partial x}\mathrm{d}V = -\int_{A_1} p\mathbf{i}\,.\,\mathbf{n}\mathrm{d}A - \int_{A_2} p\mathbf{i}\,.\,\mathbf{n}\mathrm{d}A - \int_{\Delta A_3} p\mathbf{i}\,.\,\mathbf{n}\mathrm{d}A$$

$$= pA - \left(p + \frac{\mathrm{d}p}{\mathrm{d}x}\Delta x\right)\left(A + \frac{\mathrm{d}A}{\mathrm{d}x}\Delta x\right) - \left(p + k\frac{\mathrm{d}p}{\mathrm{d}x}\Delta x\right)\left(-\frac{\mathrm{d}A}{\mathrm{d}x}\Delta x\right) \qquad \text{(E.11)}$$

The factor $k(0 < k < 1)$ in the last term accounts for the fact that the pressure on side wall area ΔA_3 takes a value between p on inlet area A_1 and $p + (\mathrm{d}p/\mathrm{d}x)\Delta x$ on outlet area A_2. Moreover, the x-component of the force due to this pressure applies in the negative x-direction and has a magnitude that is equal to the product of this intermediate pressure $p + k(\mathrm{d}p/\mathrm{d}x)\Delta x$ and the projection of side wall area ΔA_3 into the x-direction, which is $A_1 - A_2 = -(\mathrm{d}A/\mathrm{d}x)\Delta x$. Multiplying out the brackets of (E.11) and neglecting terms of order Δx^2, we obtain

$$pA - \left(p + \frac{\mathrm{d}p}{\mathrm{d}x}\Delta x\right)\left(A + \frac{\mathrm{d}A}{\mathrm{d}x}\Delta x\right) + \left(p + k\frac{\mathrm{d}p}{\mathrm{d}x}\Delta x\right)\left(\frac{\mathrm{d}A}{\mathrm{d}x}\Delta x\right)$$

$$= -A\frac{\mathrm{d}p}{\mathrm{d}x}\Delta x \qquad \text{(E.12)}$$

We place equation (E.10) on the left-hand side of the integrated x-momentum equation and equation (E.12) on the right-hand side:

$$\rho u A\frac{\mathrm{d}u}{\mathrm{d}x}\Delta x = -A\frac{\mathrm{d}p}{\mathrm{d}x}\Delta x \qquad \text{(E.13)}$$

Dividing the result by Δx yields the one-dimensional momentum equation for frictionless flow:

$$\boxed{\rho u A\frac{\mathrm{d}u}{\mathrm{d}x} = -A\frac{\mathrm{d}p}{\mathrm{d}x}} \qquad \text{(E.14)}$$

Appendix F Alternative derivation for the term (n . grad ϕA_i) in Chapter 11

As shown in Figures 11.15 and 16 (Chapter 11), using co-ordinates ξ along the line joining P and A, and η along the face of the control volume (i.e. along line joining points a and b), the term grad ϕ can be written with respect to x, y co-ordinates or n, η co-ordinates as

$$\text{grad } \phi = \frac{\partial \phi}{\partial x}\mathbf{i} + \frac{\partial \phi}{\partial y}\mathbf{j} = \frac{\partial \phi}{\partial n}\mathbf{n} + \frac{\partial \phi}{\partial \eta}\mathbf{e}_\eta \tag{F.1}$$

where $\mathbf{n}$ and $\mathbf{e}_\eta$ are unit vectors along normal and tangential directions:

$$\text{grad}\phi \,.\, \mathbf{n}A_i = \left(\frac{\partial \phi}{\partial x}\mathbf{i} + \frac{\partial \phi}{\partial y}\mathbf{j}\right) . \,\mathbf{n}A_i = \left(\frac{\partial \phi}{\partial x} . \frac{\Delta y}{A_i} - \frac{\partial \phi}{\partial y} . \frac{\Delta x}{A_i}\right)A_i \tag{F.2}$$

Using the chain rule we have

$$\frac{\partial \phi}{\partial \xi} = \frac{\partial \phi}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial \phi}{\partial y}\frac{\partial y}{\partial \xi} \tag{F.3}$$

$$\frac{\partial \phi}{\partial \eta} = \frac{\partial \phi}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial \phi}{\partial y}\frac{\partial y}{\partial \eta} \tag{F.4}$$

It is usual practice to write $\partial\phi/\partial\xi = \phi_\xi$, $\partial\phi/\partial x = \phi_x$, $\partial\phi/\partial\eta = \phi_\eta$ etc.

Equations (F.3) and (F.4) can be rewritten as

$$\phi_\xi = \phi_x x_\xi + \phi_y y_\xi \tag{F.5}$$

$$\phi_\eta = \phi_x x_\eta + \phi_y y_\eta \tag{F.6}$$

Equations (F.5) and (F.6) can be solved to obtain ϕ_x and ϕ_y. The solution is

$$\phi_x = \frac{1}{\mathcal{J}}(\phi_\xi y_\eta - \phi_\eta y_\xi) = \frac{1}{\mathcal{J}}\left(\frac{\partial \phi}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial \phi}{\partial \eta}\frac{\partial y}{\partial \xi}\right) \tag{F.7}$$

$$\phi_y = \frac{1}{\mathcal{J}}(-\phi_\xi x_\eta + \phi_\eta x_\xi) = \frac{1}{\mathcal{J}}\left(-\frac{\partial \phi}{\partial \xi}\frac{\partial x}{\partial \eta} + \frac{\partial \phi}{\partial \eta}\frac{\partial x}{\partial \xi}\right) \tag{F.8}$$

where $\mathcal{J}$ is the Jacobian given by

$$\mathcal{J} = (x_\xi y_\eta - x_\eta y_\xi) = \left(\frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi}\right) \tag{F.9}$$

In equations (F.7) and (F.8) $\partial\phi/\partial\xi$ is the gradient of ϕ along the line joining P and A, and $\partial\phi/\partial\eta$ is the gradient of ϕ along the line joining a and b (the face). Other terms like $\partial x/\partial\xi$, $\partial y/\partial\xi$, $\partial x/\partial\eta$ and $\partial y/\partial\eta$ are simple geometrical quantities. Expressions for those are given below.

By substituting (F.7) and (F.8) into (F.2) we have

$$\text{grad}\ \phi\ .\ \mathbf{n}A_f = \left(\frac{\partial\phi}{\partial x}\ .\ \frac{\Delta y}{A_f} - \frac{\partial\phi}{\partial y}\ .\ \frac{\Delta x}{A_f}\right)A_i$$

$$= \frac{1}{\mathcal{J}}\left[\left(\frac{\partial\phi}{\partial\xi}\frac{\partial y}{\partial\eta} - \frac{\partial\phi}{\partial\eta}\frac{\partial y}{\partial\xi}\right)\Delta y - \left(-\frac{\partial\phi}{\partial\xi}\frac{\partial x}{\partial\eta} + \frac{\partial\phi}{\partial\eta}\frac{\partial x}{\partial\xi}\right)\Delta x\right]$$

$$= \frac{1}{\mathcal{J}}\left[\left(\frac{\partial y}{\partial\eta}\Delta y + \frac{\partial x}{\partial\eta}\Delta x\right)\frac{\partial\phi}{\partial\xi} - \left(\frac{\partial y}{\partial\xi}\Delta y + \frac{\partial x}{\partial\xi}\Delta x\right)\frac{\partial\phi}{\partial\eta}\right] \quad \text{(F.10)}$$

The two flux gradients involved in this expression may be calculated from

$$\frac{\partial\phi}{\partial\xi} = \frac{\phi_A - \phi_P}{\Delta\xi} \quad \text{and} \quad \frac{\partial\phi}{\partial\eta} = \frac{\phi_b - \phi_a}{\Delta\eta}$$

where $\Delta\xi = d_{PA}$ is the distance between points A and P and $\Delta\eta = d_{ab}$ is the distance between vertices a and b (or in this case equal to A_i). The first term involving $\partial\phi/\partial\xi$ is called the direct gradient and the second term involving $\partial\phi/\partial\eta$ is called the cross-diffusion term.

The other geometrical quantities in equation (F.10) may be obtained from

$$\frac{\partial x}{\partial\xi} = \frac{x_A - x_P}{\Delta\xi} \quad \frac{\partial y}{\partial\xi} = \frac{y_A - y_P}{\Delta\xi}$$

$$\frac{\partial x}{\partial\eta} = \frac{x_b - x_a}{\Delta\eta} \quad \frac{\partial y}{\partial\eta} = \frac{y_b - y_a}{\Delta\eta}$$

Using Mathur and Murthy's (1997; see the references in Chapter 11) approach, expression (F.9) can be written using normal unit vector $\mathbf{n}$ and two other unit vectors $\mathbf{e}_\xi$ and $\mathbf{e}_\eta$ in the directions of ξ and η respectively.

Recall that

$$\mathbf{n} = \frac{\Delta y}{A_f}\mathbf{i} - \frac{\Delta x}{A_f}\mathbf{j} = \frac{y_b - y_a}{\Delta\eta}\mathbf{i} - \frac{x_b - x_a}{\Delta\eta}\mathbf{j} \quad \text{(F.11)}$$

The unit vector along the ξ-direction may be defined as

$$\mathbf{e}_\xi = \frac{x_A - x_P}{\Delta\xi}\mathbf{i} + \frac{y_A - y_P}{\Delta\xi}\mathbf{j} \quad \text{(F.12)}$$

The unit vector along the η-direction may be defined as

$$\mathbf{e}_\eta = \frac{x_b - x_a}{\Delta\eta}\mathbf{i} + \frac{y_b - y_a}{\Delta\eta}\mathbf{j} \quad \text{(F.13)}$$

The Jacobian in the expanded form is

$$\mathcal{J} = \left(\frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \right)$$

$$= \left(\frac{(x_A - x_P)}{\Delta \xi} \frac{(y_b - y_a)}{\Delta \eta} - \frac{(x_b - x_a)}{\Delta \eta} \frac{(y_A - y_P)}{\Delta \xi} \right)$$

$$\mathcal{J} = \mathbf{n} \,.\, \mathbf{e}_\xi \tag{F.14}$$

Using the above simplification for $\mathcal{J}$ the first term of equation (F.10) may be written as

$$\frac{1}{\mathcal{J}} \left[\left(\frac{\partial y}{\partial \eta} \Delta y + \frac{\partial x}{\partial \eta} \Delta x \right) \right] \frac{\partial \phi}{\partial \xi} = \frac{1}{\mathbf{n} \,.\, \mathbf{e}_\xi} \left[\frac{(y_b - y_a)}{\Delta \eta} (y_b - y_a) + \frac{(x_b - x_a)}{\Delta \eta} (x_b - x_a) \right] \frac{\partial \phi}{\partial \xi}$$

$$= \frac{1}{\mathbf{n} \,.\, \mathbf{e}_\xi} [\mathbf{n} \,.\, \mathbf{n} A_i] \frac{\partial \phi}{\partial \xi}$$

$$= \frac{\mathbf{n} \,.\, \mathbf{n}}{\mathbf{n} \,.\, \mathbf{e}_\xi} A_i \frac{\partial \phi}{\partial \xi} \tag{F.15}$$

Similarly the cross-diffusion term of equation (F.10) may be written as

$$-\frac{1}{\mathcal{J}} \left[\left(\frac{\partial y}{\partial \xi} \Delta y + \frac{\partial x}{\partial \xi} \Delta x \right) \right] \frac{\partial \phi}{\partial \xi} = -\frac{1}{\mathbf{n} \,.\, \mathbf{e}_\xi} \left[\frac{(y_A - y_P)}{\Delta \xi} (y_b - y_a) + \frac{(x_A - x_P)}{\Delta \xi} (x_b - x_a) \right] \frac{\partial \phi}{\partial \eta}$$

$$= -\frac{1}{\mathbf{n} \,.\, \mathbf{e}_\xi} [\mathbf{e}_\xi \,.\, \mathbf{e}_\eta \Delta \eta] \frac{\partial \phi}{\partial \eta}$$

$$= -\frac{\mathbf{e}_\xi \,.\, \mathbf{e}_\eta A_i}{\mathbf{n} \,.\, \mathbf{e}_\xi} \frac{\partial \phi}{\partial \eta} \tag{F.16}$$

The complete diffusion contribution in vector form is

$$\mathbf{n} \,.\, \mathrm{grad}\, \phi A_f = \frac{\mathbf{n} \,.\, \mathbf{n} A_i}{\mathbf{n} \,.\, \mathbf{e}_\xi} \frac{\partial \phi}{\partial \xi} - \frac{\mathbf{e}_\xi \,.\, \mathbf{e}_\eta A_i}{\mathbf{n} \,.\, \mathbf{e}_\xi} \frac{\partial \phi}{\partial \eta} \tag{F.17}$$

Appendix G Some examples

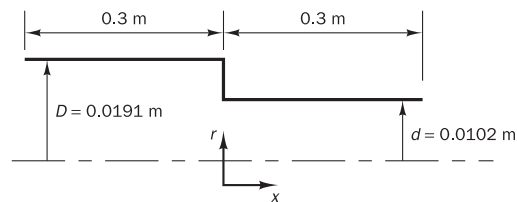
G.1 Some applications

In this appendix we give three examples of simple CFD applications. An interested reader with access to commercial or any other CFD software could repeat these exercises to gain some hands-on experience. The boundary conditions and problem specification are briefly described and specimen results are presented to illustrate how CFD and modelling of combustion can be applied to practical situations. The examples are presented without finer details of the calculations, which can be found in the cited references.

G.2 Flow in a sudden pipe contraction

The problem considered

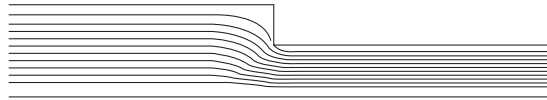
This problem was selected to illustrate the application of CFD to a benchmark problem with a set of well-documented data for the comparison of predictions with experiments. The problem considered here is laminar pipe flow in a sudden contraction shown in Figure G.1. Durst and Loy (1985) have provided the experimental data for a range of Reynolds numbers. The flow with a Reynolds number ($Re = \rho UD/\mu$) of 372 was considered for CFD modelling, where U is the average velocity in the pipe with diameter D.

Figure G.1 Experimental configuration of Durst and Loy (1985)

CFD simulation

The geometry was modelled with a two-dimensional axisymmetric grid of 100×60. The velocity profile for fully developed laminar flow was imposed at the inlet, and the no-slip condition was applied at wall boundaries. At the exit plane, all derivatives in the axial direction were set to zero. The CFD calculation was carried out using the SIMPLER algorithm and the hybrid differencing scheme.

Specimen results

Since the flow is laminar, the governing equations are exact (i.e. no turbulence modelling involved here). The predicted streamlines of the flow are shown in Figure G.2. The velocity profiles are shown in Figure G.3 for six different cross-sections of the domain: three before, and three after, the contraction. The experimental data of Durst and Loy (1985) are also included for comparison. It can be seen that the predictions agree well with the experimental measurements. Further grid refinement did not cause significant changes in the predictions, and therefore these results can be considered to be grid independent. It should be noted that comparisons for locations other than those shown in the figure and for other Reynolds numbers also agree well with the experimental data. This simple example shows the capability of CFD to predict practical flow situations, with a good degree of accuracy.

Figure G.2 Predicted streamline pattern

G.3 Modelling of a fire in a test room

The problem considered

In contrast to the previous benchmark problem we now study a case at the other end of the spectrum of complexity. We compare CFD calculations with experimental fire tests carried out by the Lawrence Livermore National Laboratory (LLNL) in the test room shown in Figure G.4. The details of the experiments have been reported in Alvarez *et al.* (1984). The fire was at the centre of the floor and clean air was introduced along the floor of the test cell, which is approximated in the model by a 0.12 m high and 2 m long slot for air entry, located 0.1 m above the floor. The fire sources in the experiments were a burner, a spray and a pool of fuel in a tray. The products of combustion were extracted near the top of the cell using an axial flow fan through a rectangular 0.65 m square duct placed 3.6 m above the floor, as shown in Figure G.4. A total of 27 tests were reported by Alvarez *et al.* (1984), and the one designated MOD08 has been selected for CFD modelling here. In this test, a spray of isopropyl alcohol from an opposed-jet nozzle located at the centre of the pan was used, and the fuel evaporated quickly to burn in a way similar to a natural pool fire. The fuel injection rate was 13.1 g/s with a total heat release rate of 400 kW. These data were used to specify burner conditions at the fire source. The measured extraction rate, 400 l/s in the steady state, was used to specify the outflow. The mass flow rate of air into the domain and the inlet and outlet velocities are calculated as part of the solution. The walls, the floor and the ceiling of the compartment were of 0.1 m thick refractory. The estimated thermal conductivity, density and specific heat were, respectively, 0.39 W/m.K, 1400 kg/m^3 and 1 kJ/kg.K for the walls and 0.63 W/m.K, 1920 kg/m^3 and 1 kJ/kg.K for the ceiling and the floor. The walls were assumed to be perfectly black for radiation calculations.

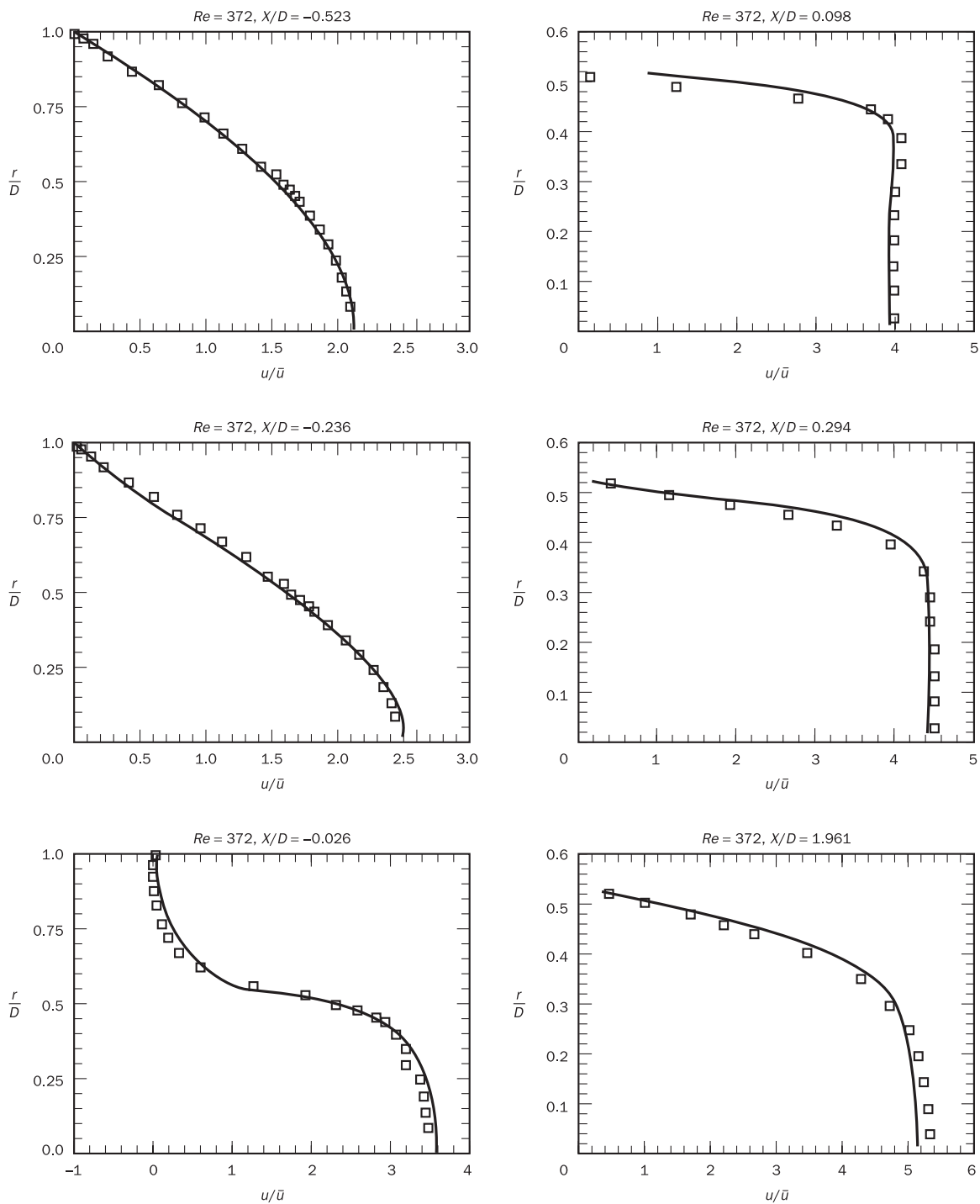
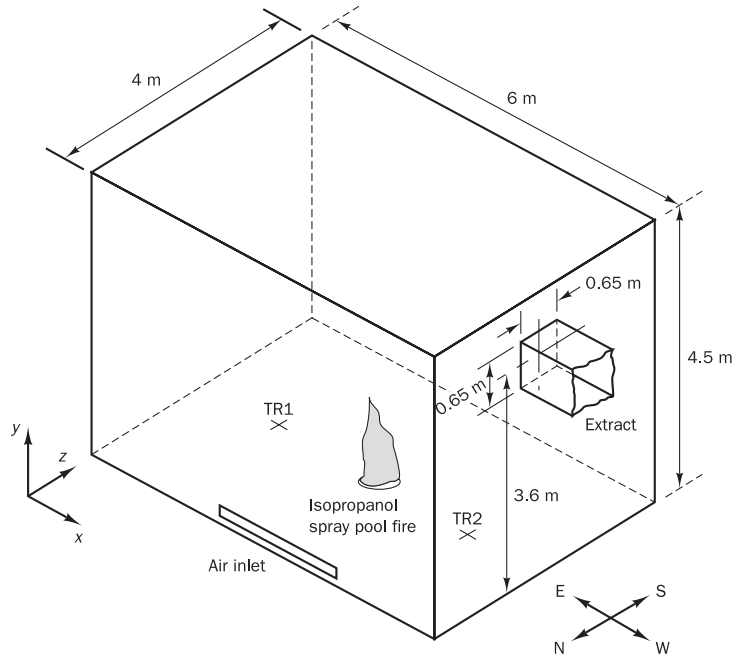



Figure G.3 Comparison of predictions and experimental results at six different locations

Figure G.4 Schematic diagram of the Lawrence Livermore National Laboratory (LLNL) fire test cell

CFD simulation

The simulation of the aerodynamics and combustion was carried out using a three-dimensional CFD procedure based on the SIMPLE algorithm and the hybrid differencing scheme for discretisation. Turbulence was modelled with the k–ε turbulence model with buoyancy terms, and combustion modelling assumed fast chemistry (SCRS). The discrete transfer model of thermal radiation (Lockwood and Shah, 1981) was used to calculate radiative heat transfer. The wall temperatures were obtained from a one-dimensional wall heat transfer model. A numerical grid of $14 \times 13 \times 12$, although not very fine, was considered adequate to predict the overall properties of the fire. Further details of the model can be found in Malalasekera (1988) and Lockwood and Malalasekera (1988). Some specimen results are presented below.

Specimen results

Figure G.5 shows the predicted steady state flow pattern in the Y–Z plane at $X = 3.25$ m. The buoyancy-generated flow is clearly reproduced by the simulation, which also shows the entrainment induced by the strong buoyancy effects. The predicted temperature distribution in the Y–Z plane at $X = 3.00$ m (Figure G.6) shows the hot gases around the central flame and the formation of a hot layer at ceiling level. The flame structure and tilt due to induced air flow are also clearly visible. Figure G.7 compares the room temperature predictions with the experimental data of Alvarez *et al.* (1984). The experimental temperatures were recorded using two thermocouple rakes (TR1, east rake; and TR2, west rake) with 15 thermocouples each placed 1.5 m on either side of the fire and located in the central plane as shown in Figure G.4.

Figure G.5 Predicted flow inside the compartment: velocity vector plots in the Z–Y plane at $X = 3.25$ m

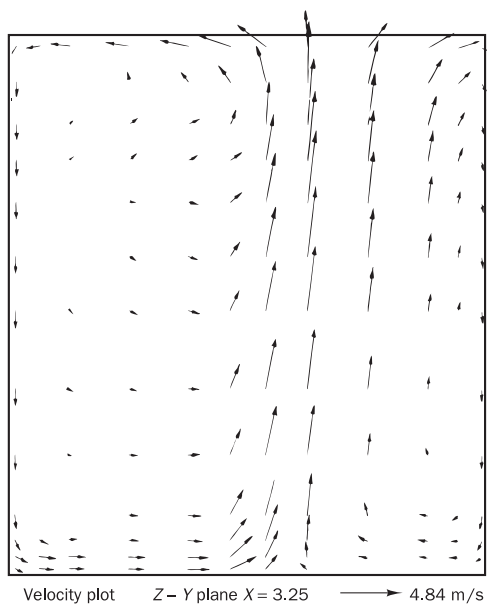



Figure G.6 Predicted temperature (K) field in the Y–Z plane at $X = 3.00$ m

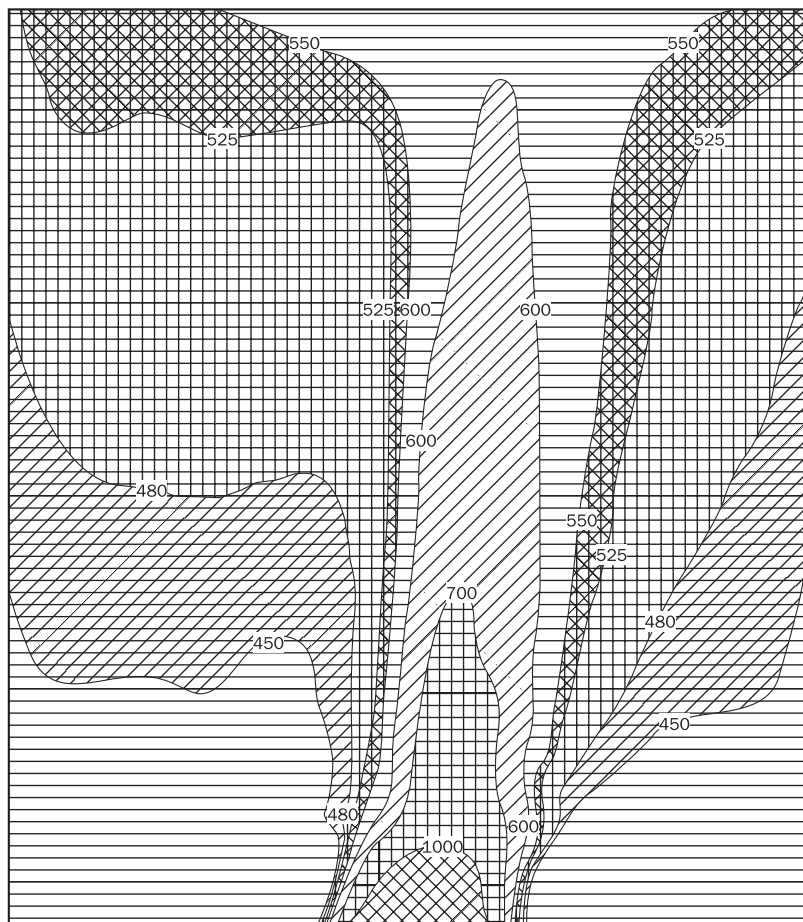
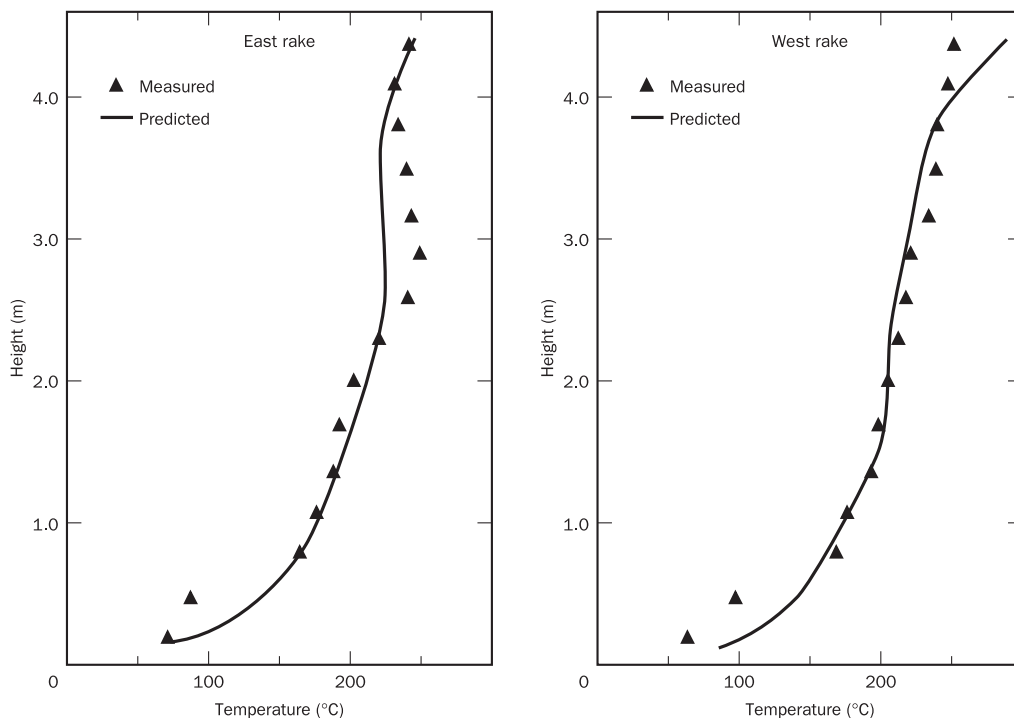



Figure G.7 Comparison of predicted and measured temperature distributions for LLNL test MOD08

The predictions and experiments show good agreement, which illustrates the capability of CFD in predicting complex flows. The predictions reproduce the main features of the experiments and, despite the coarse grid, the predictions agree well with the experimental data.

G.4 Laminar flow in a circular pipe driven by periodic pressure variations

The problem considered

Many engineering problems involve unsteady behaviour. In some cases, for instance paint mixing, the flow may be steady but the distribution of a transported scalar variable changes with time. In the present example we consider one of the simplest cases of the class of problems with genuinely unsteady flow fields: the periodic oscillations of an incompressible laminar flow in a circular pipe driven by harmonic pressure variations between inlet and outlet. Blood flows in veins and arteries, pressure waves in oil pipelines and air flows in intake manifolds of internal combustion engines can be modelled as periodic duct flows.

The applied pressure difference between the pipe ends is varied according to

$$\Delta P = K \cos nt \tag{G.1}$$

The amplitude K is taken to be 50 000 (Pa) and the circular frequency n is equal to 2π Hz, giving an oscillation period of 1 s. Schlichting (1979) gives the analytical solution for the axial velocity component $u(r, t)$ as a function of radius r and time t for periodic laminar flow in a very long pipe as the real part of the following expression:

$$u(r, t) = -i\left(\frac{K}{n\rho L}\right)e^{int}\left[1 - \frac{\mathcal{J}_0\left(r\sqrt{\frac{-in}{\nu}}\right)}{\mathcal{J}_0\left(R\sqrt{\frac{-in}{\nu}}\right)}\right] \tag{G.2}$$

In this formula ρ, ν and L are the fluid density, kinematic viscosity and the length of the pipe respectively, $\mathcal{J}_0$ denotes the Bessel function of the first kind of order 0, and i is $\sqrt{-1}$. The general features of the velocity distributions are dependent on the value of the non-dimensional parameter $\sqrt{(n/\nu)}R$. The solution behaviour for small and large values of this parameter will be discussed below. Here we calculate the flow for two intermediate values of n by taking the pipe radius R equal to 0.01 m and frequency n as 2π Hz in conjunction with a constant fluid density of 1000 kg/m^3 and dynamic viscosity of 0.4 and 0.1 kg/m.s. This yields values of $\sqrt{(n/\nu)}R$ of 1.253 and 2.507 respectively.

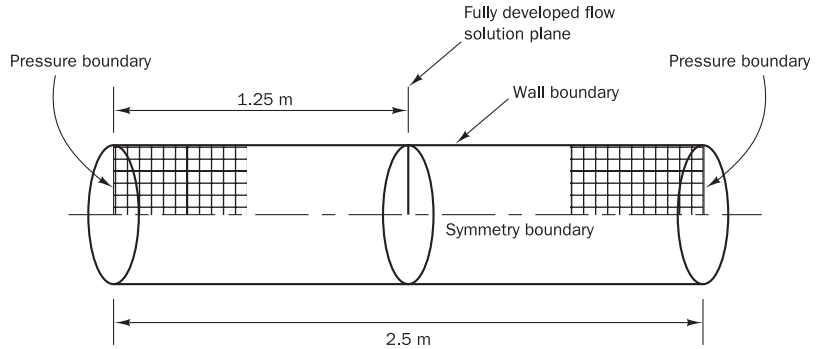
CFD simulation

In order to set up a valid comparison between the analytical and finite volume solution of this problem we need to consider a pipe of sufficient length. The boundary layer flow near the inlet of a pipe changes in the downstream direction, and in a steady flow the velocity distribution becomes fully developed after a distance l_E given by (Schlichting, 1979)

$$\frac{l_E}{R} = 0.25\frac{\bar{u}R}{\nu} \tag{G.3}$$

An estimate of the maximum possible mean velocities $\bar{u}$ (approximately 4 m/s here) can be obtained from the Hagen–Poiseuille formula (Schlichting, 1979). This leads to a maximum Reynolds number of 800 and hence a value of l_E of 1 m. Since the flow switches direction in the course of a cycle it is necessary to employ a computational domain of length greater than two times l_E to ensure that there is always a section of fully developed flow half way along the duct. In this simulation we use a domain with a length L equal to 2.5 m and consider the solution in a cross-sectional plane at a distance of 1.25 m from its ends.

The flow is axisymmetric, and we use a grid of 250 axial and 20 radial nodes distributed uniformly in the z- and r-directions. Figure G.8 shows a sketch of the solution domain and part of the mesh used. At $r = 0$ a symmetry boundary condition ensures that there is no flow across the axis and that the gradients of all variables in the radial direction are locally zero. At $r = R = 0.01$ m the usual wall boundary condition is maintained. The cosinusoidal driving pressure difference given by Figure G.9 and equation 10.36 is applied by means of prescribed pressure boundary conditions at $z = 0$ and

Figure G.8 Solution domain and a part of the mesh for simulation of periodic laminar pipe flow

$z = L = 2.5$ m. The solution procedure is SIMPLER with fully implicit time marching; the time step is 1 ms. The parabolic velocity profile of a steady laminar pipe flow is used as the initial velocity field.

Specimen results

Figures G.9 and G.10 compare the numerical and analytical solutions half way along the pipe at time intervals of 0.125 s. The finite volume solution is studied after three pressure cycles, allowing time for the initial transients to die out. It is clear from the solution that the agreement between numerical and analytical solutions is generally excellent. There are minor discrepancies in the simulation with $\sqrt{(n/\nu)}R = 2.507$ during those parts of the solution cycle where the flow near the boundary moves in the opposite direction to that in the core of the pipe. These can be explained by the fact that the local pressure gradient $\partial p/\partial x$ is somewhat different from the overall pressure gradient $\Delta p/L$ due to energy losses between the inlet and the solution cross-section.

Figure G.9 Imposed transient pressure cycle

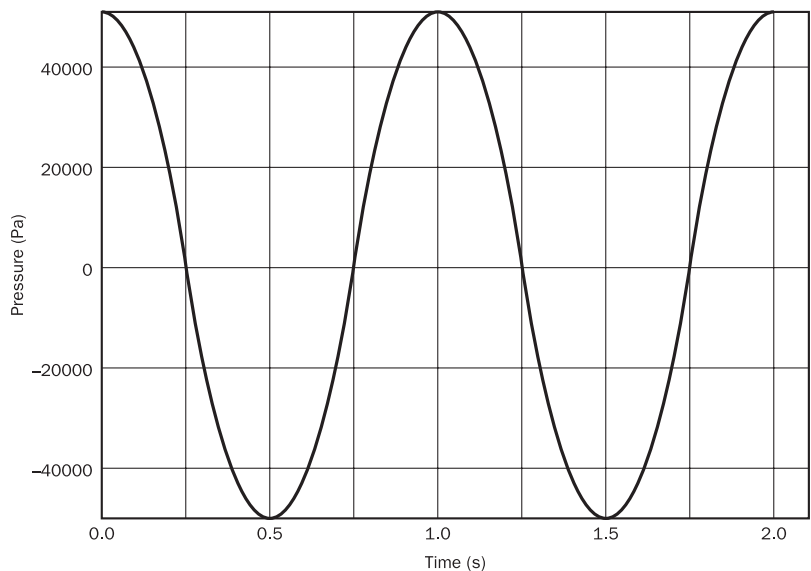
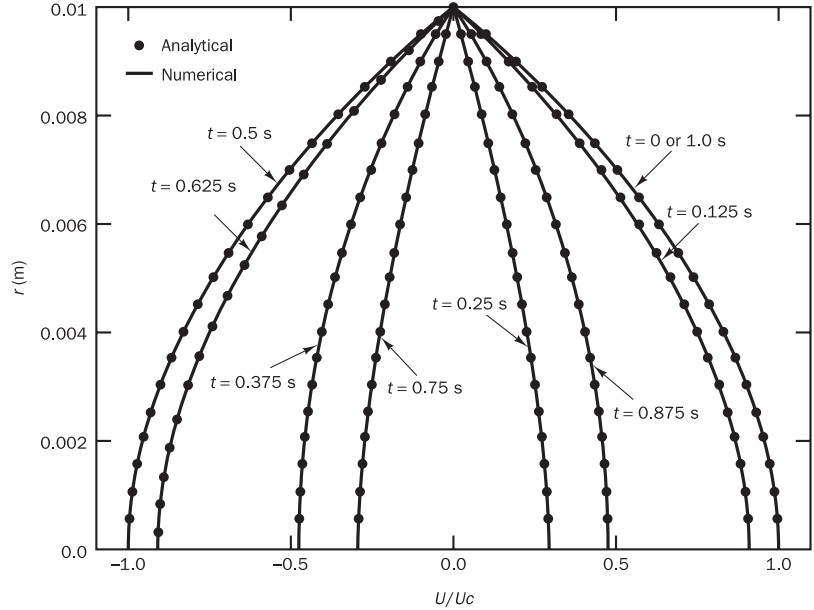



Figure G.10 Velocity distribution for periodic laminar pipe flow with $\sqrt{(n/\nu)}R = 1.253$

Overall flow behaviour can be explained by considering appropriate expressions (Abramowitz and Stegun, 1964) of the Bessel function $\mathcal{J}_0$ in the analytical solution.

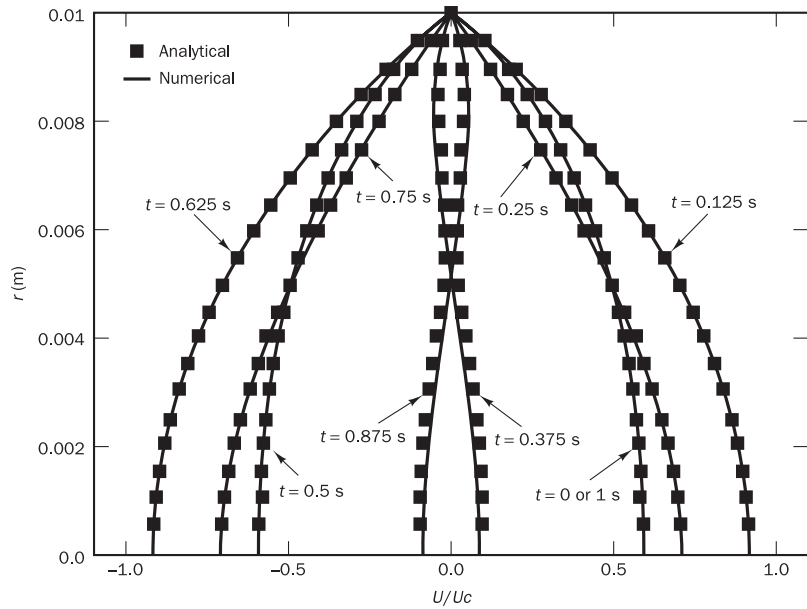
For very slow oscillations $\sqrt{(n/\nu)}R \to 0$ we obtain

$$u(r, t) = \frac{K}{4\nu}(R^2 - r^2) \cos nt \tag{G.4}$$

This exhibits the parabolic velocity distribution of a steady, fully developed, laminar pipe flow with a periodic time variation. The amplitude depends on the fluid viscosity, and the oscillations are in phase with the driving pressure difference. For fast oscillations $\sqrt{(n/\nu)}R \to \infty$ we have

$$u(r, t) = \frac{K}{n}\left[\sin nt - \sqrt{\frac{R}{r}}\exp\left(-\sqrt{\frac{n}{2\nu}}(R - r)\right)\sin\left(nt - \sqrt{\frac{n}{2\nu}}(R - r)\right)\right] \tag{G.5}$$

Expression (G.5) contains two sinusoidal terms, the first of which is independent of viscosity. It describes the flow in the central core of the pipe, which has a uniform velocity distribution with an amplitude inversely proportional to the oscillation frequency and a phase lag of $\pi/2$ radians behind the excitation force. The amplitude and the phase of the second term are viscosity dependent. The term decays quickly with distance $(R - r)$ from the pipe wall due to the exponential factor. It can be shown that this boundary layer flow lags behind the driving pressure difference by $\pi/4$ radians. The phase difference between the core and the boundary layer gives rise to an annular flow pattern during fast oscillations. It is clear that the results of Figures G.10 and G.11 exhibit the main characteristics of the slow and fast solution respectively.

Figure G.11 Velocity distribution for periodic laminar pipe flow with $\sqrt{(n/\nu)}R = 2.507$

The above flow can be comfortably calculated on a workstation, but this success should not mislead the prospective user of commercial CFD codes. Other types of unsteady flow problems with complex geometries and/or fluid physics such as turbulent intake manifold flows (Chen, 1994), pulsed combustion (Benelli *et al.*, 1992), transient free convective cooling of warm crude oil in storage tanks (Cotter and Charles, 1993) or hydrodynamic instabilities such as vortex shedding require very large computing resources. Often such flow calculations are only practical within reasonable time limits on dedicated large computers with advanced architecture and specially adapted algorithm structures.

Bibliography

Chapter 1

Gottlieb, D. and Orszag, S. A. (1977). *Numerical Analysis of Spectral Methods: Theory and Applications*, SIAM, Philadelphia.

Hastings, C. (1985). *Approximations for Digital Computers*, Princeton University Press, Princeton, NJ.

Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Smith, G. D. (1985). *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 3rd edn, Clarendon Press, Oxford.

Zienkiewicz, O. C. and Taylor, R. L. (1991). *The Finite Element Method – Vol. 2: Solid and Fluid Mechanics*, McGraw-Hill, New York.

Chapter 2

Bland, D. R. (1988). *Wave Theory and Applications*, Clarendon Press, Oxford.

Fletcher, C. A. J. (1991). *Computational Techniques for Fluid Dynamics*, Vols I and II, Springer-Verlag, Berlin.

Gresho, P. M. (1991). Incompressible Fluid Dynamics: Some Fundamental Formulation Issues, *Ann. Rev. Fluid Mech.*, Vol. 23, pp. 413–453.

Issa, R. I. and Lockwood, F. C. (1977). On the Prediction of Two-dimensional Supersonic Viscous Interactions near Walls, *AIAA J.*, Vol. 15, No. 2, pp. 182–188.

McGuirk, J. J. and Page, G. J. (1990). Shock Capturing Using a Pressure-Correction Method, *AIAA J.*, Vol. 28, No. 10, pp. 1751–1757.

Schlichting, H. (1979). *Boundary-layer Theory*, 7th edn, McGraw-Hill, New York.

Shapiro, A. H. (1953). *Compressible Fluid Flow*, Vol. 1, John Wiley & Sons, New York.

The Open University (1984). *Mathematical Methods and Fluid Mechanics*, Course MST322, The Open University Press, Milton Keynes.

Chapter 3

Abbott, M. B. and Basco, D. R. (1989). *Computational Fluid Dynamics – An Introduction for Engineers*, Longman Scientific & Technical, Harlow.

Anderson, D. A., Tannehill, J. C. and Pletcher, R. H. (1984). *Computational Fluid Mechanics and Heat Transfer*, Hemisphere, New York.

Bardina, J., Ferziger, J. H. and Reynolds, W. C. (1980). Improved Subgrid-scale Models for Large-eddy Simulation, AIAA Paper 80-1357.

Bradshaw, P., Cebeci, T. and Whitelaw, J. H. (1981). *Engineering Calculation Methods for Turbulent Flow*, Academic Press, London.

Buchhave, P., George, W. K. Jr and Lumley, J. L. (1979). The Measurement of Turbulence with the Laser-Doppler Anemometer, *Ann. Rev. Fluid Mech.*, Vol. 11, pp. 443–503.

Cebeci, T. (1989). Essential Ingredients of a Method for Low Reynolds-number Airfoils, *AIAA J.*, Vol. 27, No. 12, pp. 1680–1688.

Champagne, F. H., Pao, Y. H. and Wygnanski, I. J. (1976). On the Two-dimensional Mixing Region, *J. Fluid Mech.*, Vol. 74, Pt 2, pp. 209–250.

Chen, H.-C. and Patel, V. C. (1988). Near-wall Turbulence Models for Complex Flows Including Separation, *AIAA J.*, Vol. 26, No. 6, pp. 641–648.

Clark, R. A., Ferziger, J. H. and Reynolds, W. C. (1979). Evaluation of Subgrid-scale Models Using an Accurately Simulated Turbulent Flow, *J. Fluid Mech.*, Vol. 91, pp. 1–16.

Comte-Bellot, G. (1976). Hot-wire Anemometry, *Ann. Rev. Fluid Mech.*, Vol. 8, pp. 209–231.

Craft, T. J., Launder, B. E. and Suga, K. (1996). Development and Application of a Cubic Eddy Viscosity Model of Turbulence, *Int. J. Heat Fluid Flow*, Vol. 17, pp. 108–115.

Deardorff, J. W. (1970). A Numerical Study of Three-dimensional Turbulent Channel Flow at Large Reynolds Numbers, *J. Fluid Mech.*, Vol. 41, pp. 453–480.

Deardorff, J. W. (1973). The Use of Subgrid Transport Equations in a Three-dimensional Model of Atmospheric Turbulence, *Trans. ASME*, *J. Fluids Eng.*, Vol. 95, Ser. I, pp. 429–438.

Demuren, A. O. and Rodi, W. (1984). Calculation of Turbulence-driven Secondary Motion in Non-circular Ducts, *J. Fluid Mech.*, Vol. 140, pp. 189–222.

Elghobashi, S. and Truesdell, G. C. (1993). Two-way Interaction between Homogeneous Turbulence and Dispersed Solid Particles. I. Turbulence Modification, *Phys. Fluids A*, Vol. 5, No. 7, pp. 1790–1801.

Evangelinos, C., Lucor, D. and Karniadakis, G. E. (2000). DNS-derived Force Distribution on Flexible Cylinders Subject to Vortex-induced Vibration, *J. Fluids Struct.*, Vol. 14, No. 3, pp. 429–440.

Ferrante, A. and Elgobashi, S. E. (2004). A Robust Method for Generating Inflow Conditions for Direct Simulations of Spatially-developing Turbulent Boundary Layers, *J. Comput. Phys.*, Vol. 198, pp. 372–387.

Ferziger, J. H. (1977). Large Eddy Numerical Simulations of Turbulent Flows, *AIAA J.*, Vol. 15, No. 9, pp. 1261–1267.

Fureby, C., Tabor, G., Weller, H. G. and Gosman, A. D. (1997). A Comparative Study of Subgrid Scale Models in Homogeneous Isotropic Turbulence, *Phys. Fluids*, Vol. 9, No. 5, pp. 1416–1429.

Germano, M. (1986). A Proposal for a Redefinition of the Turbulent Stresses in the Filtered Navier–Stokes Equations, *Phys. Fluids*, Vol. 29, pp. 2323–2324.

Germano, M., Piomelli, U., Moin, P. and Cabot, W. H. (1991). A Dynamic Subgrid-scale Eddy Viscosity Model, *Phys. Fluids A*, Vol. 3, pp. 1760–1765.

Geurts, B. J. and Leonard, A. (2005). Is LES Ready for Complex Flows? Preprint http://www.newton.cam.ac.uk/preprints/NI99009.pdf.

Ghosal, S. and Moin, P. (1995). The Basic Equations for the Large Eddy Simulation of Turbulent Flows in Complex Geometry, *J. Comput. Phys.*, Vol. 118, pp. 24–37.

Gutmark, E. and Wygnanski, I. (1976). The Planar Turbulent Jet, *J. Fluid Mech.*, Vol. 73, Pt 3, pp. 465–495.

Hanjalič, K. (2004). *Closure Models for Incompressible Turbulent Flows*, Von Karman Institute Lecture Series Turbulence 2004, Von Karman Institute, Rhode-Saint Genese, Belgium, http://www.vki.ac.be/educat/lect-ser/2004/turbulence2004/hanjalic.pdf.

Hoarau, Y., Faghani, D., Braza, M., Perrin, R., Anne-Archard, D. and Ruiz, D. (2003). Direct Numerical Simulation of the Three-dimensional Transition to Turbulence in the Incompressible Flow around a Wing, *Flow, Turbul. Combust.*, Vol. 71, No. 1–4, pp. 119–132.

Horiuti, K. (1990). Higher-order Terms in the Anisotropic Representation of Reynolds Stresses, *Phys. Fluids A*, Vol. 2, No. 10, pp. 1708–1710.

Jiang, X. and Luo, K. H. (2000). Spatial Direct Numerical Simulation of the Large Vortical Structures in Forced Plumes, *Flow, Turbul. Combust.*, Vol. 64, No. 1, pp. 43–69.

Jones, W. P. and Whitelaw, J. H. (1982). Calculation of Turbulent Reacting Flows: A Review, *Combust. Flame*, Vol. 48, pp. 1–26.

Karniadakis, G. E. (1989). Spectral Element Simulations of Laminar and Turbulent Flows in Complex Geometries, *Appl. Num. Math.*, Vol. 6, No. 1–2, pp. 85–105.

Karniadakis, G. E. (1990). Spectral Element-Fourier Methods for Incompressible Turbulent Flows, *Comput. Methods Appl. Mech. Eng.*, Vol. 80, No. 1–3, pp. 367–380.

Kasagi, N. (1998). Progress in Direct Numerical Simulation of Turbulent Transport and its Control, *Int. J. Heat Fluid Flow*, Vol. 19, pp. 125–134.

Klebanoff, P. S. (1955). *Characteristics of Turbulence in a Boundary Layer with Zero Pressure Gradient*, NACA Report 1247, National Bureau of Standards, Washington, DC.

Klein, M., Sadiki, A. and Janicka, J. (2003). A Digital Filter Based Generation of Inflow Data for Spatially Developing Direct Numerical or Large Eddy Simulations, *J. Comput. Phys.*, Vol. 186, pp. 652–665.

Kleiser, L. and Zang, T. A. (1991). Numerical Simulation of Transition in Wall-bounded Shear Flows, *Ann. Rev. Fluid Mech.*, Vol. 23, pp. 495–537.

Lam, C. K. G. and Bremhorst, K. A. (1981). Modified Form of the k–ε Model for Predicting Wall Turbulence, *J. Fluids Eng.*, Vol. 103, pp. 456–460.

Laufer, J. (1952). *The Structure of Turbulence in Fully Developed Pipe Flow*, NACA Report 1174, National Bureau of Standards, Washington, DC.

Launder, B. E. (1989). Second-moment Closures: Present and Future?, *Int. J. Heat Fluid Flow*, Vol. 10, pp. 282–300.

Launder, B. E. and Sharma, B. I. (1974). Application of the Energy-Dissipation Model of Turbulence to the Calculation of Flow near a Spinning Disc, *Lett. Heat Mass Transfer*, Vol. 1, pp. 131–137.

Launder, B. E. and Spalding, D. B. (1974). The Numerical Computation of Turbulent Flows, *Comput. Methods Appl. Mech. Eng.*, Vol. 3, pp. 269–289.

Launder, B. E., Reece, G. J. and Rodi, W. (1975). Progress in the Development of a Reynolds-stress Turbulence Closure, *J. Fluid Mech.*, Vol. 68, Pt 3, pp. 537–566.

Lele, S. K. (1997). Computational Aeroacoustics: A Review, AIAA Paper 97-0018.

Leonard, A. (1974). Energy Cascade in Large-eddy Simulations of Turbulent Fluid Flows, *Adv. Geophys.*, Vol. 18A, pp. 237–248.

Lesieur, M. and Métais, O. (1996). New Trends in Large-eddy Simulations of Turbulence, *Ann. Rev. Fluid Mech.*, Vol. 28, pp. 45–82.

Leslie, D. C. and Quarini, G. L. (1979). The Application of Turbulence Theory to the Formulation of Subgrid Modelling Procedures, *J. Fluid Mech.*, Vol. 91, pp. 65–91.

Lilly, D. K. (1966). *On the Application of the Eddy Viscosity Concept in the Inertial Sub-range of Turbulence*, NCAR Report No. 123.

Lilly, D. K. (1967). The Representation of Small-scale Turbulence in Numerical Simulation Experiments, *Proceedings of the IBM Scientific Computing Symposium on Environmental Science*, p. 195.

Lilly, D. K. (1992). A Proposed Modification of the Germano Subgrid-scale Closure Method, *Phys. Fluids A*, Vol. 4, pp. 633–635.

Lumley, J. L. (1970). Toward a Turbulent Constitutive Equation, *J. Fluid Mech.*, Vol. 41, pp. 413–434.

Lumley, J. L. (1978). Computational Modelling of Turbulent Flows, *Adv. Appl. Mech.*, Vol. 18, pp. 123–176.

Lumley, J. L. (1989). *Whither Turbulence? Turbulence at the Crossroads*, Lecture Notes in Physics No. 357, Springer-Verlag, Berlin.

Lund, T. S., Wu, X. and Squires, K. D. (1998). Generation of Turbulent Inflow Data for Spatially-developing Boundary Layer Simulations, *J. Comput. Phys.*, Vol. 140, pp. 233–258.

Luo, K. H., Bellan, J., Delichatsios, M. and Nathan, G. S. (2005). Axis Switching in Turbulent Buoyant Diffusion Flames, *Proc. Combust. Inst.*, Vol. 30, No. 1, pp. 603–610.

Marsden, A. L., Vasilyev, O. V. and Moin, P. (2002). Construction of Commutative Filters for LES on Unstructured Meshes, *J. Comput. Phys.*, Vol. 175, pp. 584–603.

McMillan, O. J. and Ferziger, J. H. (1979). Direct Testing of Subgrid-scale Models, *AIAA J.*, Vol. 17, pp. 1340–1346.

Meneveau, C. and Katz, J. (2000). Scale-invariance and Turbulence Models for Large-eddy Simulation, *Ann. Rev. Fluid Mech.*, Vol. 32, pp. 1–32.

Menter, F. R. (1992a). Performance of Popular Turbulence Models for Attached and Separated Adverse Pressure Gradient Flow, *AIAA J.*, Vol. 30, pp. 2066–2072.

Menter, F. R. (1992b). Improved Two-equation k–ω Turbulence Models for Aerodynamic Flows, NASA Technical Memorandum TM-103975, NASA Ames, CA.

Menter, F. (1994). Two-equation Eddy-viscosity Turbulence Model for Engineering Applications, *AIAA J.*, Vol. 32, pp. 1598–1605.

Menter, F. (1997). Eddy-viscosity Transport Equations and their Relation to the k–ε Model, *Trans. ASME, J. Fluids Eng.*, Vol. 119, pp. 876–884.

Menter, F. R., Kuntz, M. and Langtry, R. (2003). Ten Years of Industrial Experience with the SST Turbulence Model, *Proceedings of the Fourth International Symposium on Turbulence, Heat and Mass Transfer*, Begell House, Redding, CT.

Moin, P. (1991). Towards Large Eddy and Direct Simulation of Complex Turbulent Flows, *Comput. Methods Appl. Mech. Eng.*, Vol. 87, No. 2–3, pp. 329–334.

Moin, P. (2002). Advances in Large Eddy Simulation Methodology for Complex Flows, *Int. J. Heat Fluid Flow*, Vol. 23, pp. 710–720.

Moin, P. and Kim, J. (1997). Tackling Turbulence with Supercomputers, *Sci. Am.*, Vol. 276, No. 1, pp. 62–68.

Moin, P. and Mahesh, K. (1998). Direct Numerical Simulation: A Tool in Turbulence Research, *Ann. Rev. Fluid Mech.*, Vol. 30, pp. 539–578.

Monin, A. S. and Yaglom, A. M. (1971). *Statistical Fluid Mechanics: Mechanics of Turbulence*, Vol. 1, MIT Press, Cambridge, MA.

Nakayama, Y. (ed.) (1988). *Visualised Flow*, Pergamon Press, Oxford.

Naot, D. and Rodi, W. (1982). Numerical Simulation of Secondary Currents in Channel Flow, *J. Hydraul. Div. ASCE*, Vol. 108 (HY8), pp. 948–968.

Orlandi, P. and Fatica, M. (1997). Direct Simulations of Turbulent Flow in a Pipe Rotating about its Axis, *J. Fluid Mech.*, Vol. 343, pp. 43–72.

Orszag, S. A. and Patera, A. T. (1984). A Spectral Element Method for Fluid Dynamics: Laminar Flow in a Channel Expansion, *J. Comput. Phys.*, Vol. 54, p. 468.

Orszag, S. A. and Patterson, G. S. (1972). Numerical Simulation of Three-dimensional Homogeneous Isotropic Turbulence, *Phys. Rev. Lett.*, Vol. 28, pp. 76–79.

Patel, V. C., Rodi, W. and Scheuerer, G. (1985). Turbulence Models for Near-wall and Low Reynolds Number Flows: A Review, *AIAA J.*, Vol. 23, No. 9, pp. 1308–1319.

Patera, A. T. (1986). Advances and Future Directions of Research on Spectral Methods, *Proceedings of Computational Mechanics – Advances and Trends*. Presented at the Winter Annual Meeting of the American Society of Mechanical Engineers, AMD Vol. 75, pp. 411–427.

Peyret, R. and Krause, E. (eds) (2000). *Advanced Turbulent Flow Computations*, CISM Courses and Lectures No. 395, International Centre for Mechanical Sciences, Springer-Verlag, Vienna.

Poinsot, T. J., Haworth, D. C. and Bruneaux, G. (1993). Direct Simulation and Modeling of Flame-Wall Interaction for Premixed Turbulent Combustion, *Combust. Flame*, Vol. 95, pp. 118–132.

Pope, S. B. (1975). A More General Effective Viscosity Hypothesis, *J. Fluid Mech.*, Vol. 72, pp. 331–340.

Rivlin, R. S. (1957). The Relation between the Flow of Non-Newtonian Fluids and Turbulent Newtonian Fluids, *Q. Appl. Math.*, Vol. 15, pp. 212–215.

Rodi, W. (1980). *Turbulence Models and their Application in Hydraulics – A State of the Art Review*, IAHR, Delft, The Netherlands.

Rodi, W. (1991). Experience with Two-layer Models Combining the k–ε Model with a One-equation Model near the Wall, AIAA Paper 91-0216.

Rogallo, R. S. and Moin, P. (1984). Numerical Simulation of Turbulent Flows, *Ann. Rev. Fluid Mech.*, Vol. 16, pp. 99–137.

Rogers, M. M. (2002). The Evolution of Strained Turbulent Plane Wakes, *J. Fluid Mech.*, Vol. 463, pp. 53–120.

Schlichting, H. (1979). *Boundary-layer Theory*, 7th edn, McGraw-Hill, New York.

Schumann, U. (1975). Subgrid Scale Model for Finite Difference Simulations of Turbulent Flows in Plane Channels and Annuli, *J. Comput. Phys.*, Vol. 18, pp. 376–404.

Scotti, A., Meneveau, C. and Lilly, D. K. (1993). Generalized Smagorinsky Model for Anisotropic Grids, *Phys. Fluids A*, Vol. 5, pp. 1229–1248.

Serre, E., Bountoux, P. and Launder, B. E. (2002). Direct Numerical Simulation of Transitional Turbulent Flow in a Closed Rotor-stator Cavity, *Flow, Turbul. Combust.*, Vol. 69, No. 1S, pp. 35–50.

Serre, E., Tuliszka-Sznitko, E. and Bountoux, P. (2004). Coupled Numerical and Theoretical Study of the Flow Transition between a Rotating and a Stationary Disk, *Phys. Fluids*, Vol. 16, No. 3, pp. 688–706.

Shih, T.-H., Liou, W. W., Shabbir, A., Yang, Z. and Zhu, J. (1995). A New k–ε Eddy-viscosity Model for High Reynolds Number Turbulent Flows – Model Development and Validation, *Comput. Fluids*, Vol. 24, No. 3, pp. 227–238.

Smagorinsky, J. (1963). General Circulation Experiments with the Primitive Equations. I. The Basic Experiment, *Mon. Weather Rev.*, Vol. 91, No. 3, pp. 99–164.

So, R. M. C., Lai, Y. G., Zhang, H. S. and Hwang, B. C. (1991). Second-order Near-wall Turbulence Closures: A Review, *AIAA J.*, Vol. 29, No. 11, pp. 1819–1835.

Spalart, P. and Allmaras, S. A. (1992). One-Equation Turbulence Model for Aerodynamic Flows, AIAA Paper 92-0439.

Speziale, C. G. (1987). On Non-linear k–l and k–ε Models of Turbulence, *J. Fluid Mech.*, Vol. 178, pp. 459–475.

Speziale, C. G. (1991). Analytical Methods for the Development of Reynolds-stress Closures in Turbulence, *Ann. Rev. Fluid Mech.*, Vol. 23, pp. 107–157.

Tam, C. K. W. (1995). Computational Aeroacoustics, AIAA Paper 95-0677.

Tamura, T., Miyagi, T. and Kitagishi, T. (1998). Numerical Prediction of Unsteady Pressures on a Square Cylinder with Various Corner Shapes, *J. Wind Eng. Ind. Aerodyn.*, Vol. 74–76, pp. 531–542.

Tennekes, H. and Lumley, J. L. (1972). *A First Course in Turbulence*, MIT Press, Cambridge, MA.

Tritton, D. J. (1977). *Physical Fluid Dynamics*, Van Nostrand Reinhold, Wokingham.

Van Dyke, M. (1982). *An Album of Fluid Motion*, Parabolic Press, Stanford, CA.

Vasilyev, O. V., Lund, T. S. and Moin, P. (1998). A General Class of Commutative Filters for LES in Complex Geometries, *J. Comput. Phys.*, Vol. 146, pp. 82–104.

Verstappen, R. W. C. P. and Veldman, A. E. P. (1997). Direct Numerical Simulation of Turbulence at Lower Costs, *J. Eng. Math.*, Vol. 32, pp. 143–159.

White, F. M. (1991). *Viscous Fluid Flow*, 2nd edn, McGraw-Hill, New York.

Wilcox, D. C. (1988). Reassessment of the Scale-determining Equation for Advanced Turbulence Models, *AIAA J.*, Vol. 26, No. 11, pp. 1299–1310.

Wilcox, D. C. (1993a). Comparison of Two-equation Turbulence Models for Boundary Layers with Pressure Gradients, *AIAA J.*, Vol. 31, No. 8, pp. 1414–1421.

Wilcox, D. C. (1993b). *Turbulence Modelling for CFD*, DCW Industries Inc., La Canada, CA.

Wilcox, D. C. (1994). Simulating Transition with a Two-equation Turbulence Model, *AIAA J.*, Vol. 32, pp. 247–255.

Wygnanski, I., Champagne, F. and Marasli, B. (1986). On the Large-scale Structures in Two-dimensional, Small-deficit, Turbulent Wakes, *J. Fluid Mech.*, Vol. 168, pp. 31–71.

Yakhot, V., Orszag, S. A., Thangam, S., Gatski, T. B. and Speziale, C. G. (1992). Development of Turbulence Models for Shear Flows by a Double Expansion Technique, *Phys. Fluids A*, Vol. 4, No. 7, pp. 1510–1520.

Chapter 4

MATLAB (1992). *The Student Edition of MATLAB*, The Math Works Inc., Prentice Hall, Englewood Cliffs, NJ.

Chapter 5

Boris, J. P. and Book, D. L. (1973). Flux Corrected Transport I, SHASTA, A Fluid Transport Algorithm that Works, *J. Comput. Phys.*, Vol. 11, pp. 38–69.

Boris, J. P. and Book, D. L. (1976). Solution of the Continuity Equation by the Method of Flux Corrected Transport, *J. Comput. Phys.*, Vol. 16, pp. 85–129.

Darwish, M. S. and Moukalled, F. (2003). TVD Schemes for Unstructured Grids, *Int. J. Heat Mass Transfer*, Vol. 46, pp. 599–611.

FLUENT documentation (2006). Fluent Inc., USA.

Han, T., Humphrey, J. A. C. and Launder, B. E. (1981). A Comparison of Hybrid and Quadratic-Upstream Differencing in High Reynolds Number Elliptic Flows, *Comput. Methods. Appl. Mech. Eng.*, Vol. 29, pp. 81–95.

Harten, A. (1983). High Resolution Schemes for Hyperbolic Conservation Laws, *J. Comput. Phys.*, Vol. 49, pp. 357–393.

Harten, A. (1984). On a Class of High Resolution Total-variation-stable Finite-difference Schemes, *SIAM J. Numer. Anal.*, Vol. 21, No. 1, pp. 1–23.

Hayase, T., Humphrey, J. A. C. and Greif, R. (1992). A Consistently Formulated QUICK Scheme for Fast and Stable Convergence Using Finite-volume Iterative Calculation Procedures, *J. Comput. Phys.*, Vol. 98, pp. 108–118.

Huang, P. G., Launder, B. E. and Leschziner, M. A. (1985). Discretisation of Non-linear Convection Processes: A Broad-range Comparison of Four Schemes, *Comput. Methods. Appl. Mech. Eng.*, Vol. 48, pp. 1–24.

Leonard, B. P. (1979). A Stable and Accurate Convective Modelling Procedure Based on Quadratic Upstream Interpolation, *Comput. Methods Appl. Mech. Eng.*, Vol. 19, pp. 59–98.

Leonard, B. P. (1988). Simple High-accuracy Resolution Program for Convective Modelling of Discontinuities, *Int. J. Numer. Methods Fluids*, Vol. 8, pp. 1291–1318.

Leschziner, M. A. (1980). Practical Evaluation of Three Finite Difference Schemes for the Computation of Steady-state Recirculating Flows, *Comput. Methods Appl. Mech. Eng.*, Vol. 23, pp. 293–312.

Lien, F. S. and Leschziner, M. A. (1993). Upstream Monotonic Interpolation for Scalar Transport with Application to Complex Turbulent Flows, *Int. J. Numer. Methods Fluids*, Vol. 19, pp. 527–548.

Osher, S. and Chakravarthy, S. (1984). High Resolution Schemes and the Entropy Condition, *SIAM J. Numer. Anal.*, Vol. 21, pp. 955–984.

Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Pollard, A. and Siu, A. L. W. (1982). The Calculation of Some Laminar Flows Using Various Discretisation Schemes, *Comput. Methods Appl. Mech. Eng.*, Vol. 35, pp. 293–313.

Roache, P. J. (1976). *Computational Fluid Dynamics*, Hermosa, Albuquerque, NM.

Roe, P. L. (1985). *Some Contributions to the Modelling of Discontinuous Flows*, Lectures in Applied Mechanics, Vol. 22, Springer-Verlag, Berlin, pp. 163–193.

Scarborough, J. B. (1958). *Numerical Mathematical Analysis*, 4th edn, Johns Hopkins University Press, Baltimore, MD.

Spalding, D. B. (1972). A Novel Finite-difference Formulation for Differential Expressions Involving both First and Second Derivatives, *Int. J. Numer. Methods Eng.*, Vol. 4, p. 551.

Sweby, P. K. (1984). High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws, *SIAM J. Numer. Anal.*, Vol. 21, No. 5, pp. 995–1011.

Van Albada, G. D., Van Leer, B. and Roberts, W. W. (1982). A Comparative Study of Computational Methods in Cosmic Gas Dynamics, *Astron. Astrophys.*, Vol. 108, pp. 76–84.

Van Leer, B. (1973). *Towards the Ultimate Conservative Difference Scheme I. The Quest of Monotinicity*, Lecture Notes in Physics, Vol. 18, Springer-Verlag, Berlin, pp. 163–168.

Van Leer, B. (1974). Towards the Ultimate Conservative Difference Scheme II. Monotinicity and Conservation Combined in a Second-Order Scheme, *J. Comput. Phys.*, Vol. 14, pp. 361–370.

Van Leer, B. (1977a). Towards the Ultimate Conservative Difference Scheme III: Upstream Centred Finite-difference Scheme for Ideal Compressible Flow, *J. Comput. Phys.*, Vol. 23, pp. 263–275.

Van Leer, B. (1977b). Towards the Ultimate Conservative Difference Scheme IV: A New Approach to Numerical Convection, *J. Comput. Phys.*, Vol. 23, pp. 276–299.

Van Leer, B. (1979). Towards the Ultimate Conservative Difference Scheme V: A Second-Order Sequel to Godunov's Method, *J. Comput. Phys.*, Vol. 32, pp. 101–136.

Chapter 6

Anderson, D. A., Tannehill, J. C. and Pletcher, R. H. (1984). *Computational Fluid Mechanics and Heat Transfer*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Harlow, F. H. and Welch, J. E. (1965). Numerical Calculation of Time-dependent Viscous Incompressible Flow of Fluid with Free Surface, *Phys. Fluids*, Vol. 8, pp. 2182–2189.

Issa, R. I. (1986). Solution of the Implicitly Discretised Fluid Flow Equations by Operator-Splitting, *J. Comput. Phys.*, Vol. 62, pp. 40–65.

Issa, R. I., Gosman, A. D. and Watkins, A. P. (1986). The Computation of Compressible and Incompressible Recirculating Flows, *J. Comput. Phys.*, Vol. 62, pp. 66–82.

Jang, D. S., Jetli, R. and Acharya, S. (1986). Comparison of the PISO, SIMPLER, and SIMPLEC Algorithms for the Treatment of the Pressure-Velocity Coupling in Steady Flow Problems, *Numer. Heat Transfer*, Vol. 19, pp. 209–228.

Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Patankar, S. V. and Spalding, D. B. (1972). A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-dimensional Parabolic Flows, *Int. J. Heat Mass Transfer*, Vol. 15, p. 1787.

Van Doormal, J. P. and Raithby, G. D. (1984). Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows, *Numer. Heat Transfer*, Vol. 7, pp. 147–163.

Chapter 7

Anderson, D. A., Tannehill, J. C. and Pletcher, R. H. (1984). *Computational Fluid Mechanics and Heat Transfer*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Briggs, W. L. (1987). *A Multigrid Tutorial*, 2nd edn, SIAM Publications, also see http://www.llnl.gov/casc/people/henson/mgtut/welcome.html.

Concus, P., Golub, G. H. and O'Leary, D. P. (1976). A Generalised Conjugate Gradient Method for the Numerical Solution of Elliptic Partial Differential Equations, in J. R. Bunch and D. J. Rose (eds) *Sparse Matrix Computations*, Academic Press, New York, pp. 309–332.

Fletcher, C. A. J. (1991). *Computational Techniques for Fluid Dynamics*, Vols I and II, Springer-Verlag, Berlin.

Hestenes, M. R. and Stiefel, E. L. (1952). Methods of Conjugate Gradient for Solving Linear Systems, *J. Res. NBS*, Vol. 49, pp. 409–436.

Hutchinson, B. R. and Raithby, G. D. (1986). A Multigrid Method Based on the Additive Correction Strategy, *Numer. Heat Transfer*, Vol. 9, pp. 511–537.

Kershaw, D. S. (1978). The Incomplete Cholesky Conjugate Gradient Method for the Iterative Solution of Linear Equations, *J. Comput. Phys.*, Vol. 26, pp. 43–65.

Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T. (1993). *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge University Press, Cambridge.

Ralston, A. and Rabinowitz, P. (1978). *A First Course in Numerical Analysis*, 2nd edn, International Student Edition, McGraw-Hill, Tokyo.

Reid, J. K. (1971). On the Method of Conjugate Gradients for the Solution of Large, Sparse Systems of Linear Equations, in J. K. Reid (ed.) *Large Sparse Sets of Linear Equations*, Academic Press, New York.

Schneider, G. E. and Zedan, M. (1981). A Modified Strongly Implicit Procedure for the Numerical Solution of Field Problems, *Numer. Heat Transfer, Part B*, Vol. 4, pp. 1–19.

Stone, H. L. (1968). Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations, *SIAM J. Numer. Anal.*, Vol. 5, pp. 530–559.

Thomas, L. H. (1949). *Elliptic Problems in Linear Differential Equations over a Network*, Watson Sci. Comput. Lab Report, Columbia University, New York.

Wesseling, P. (1992). *An Introduction to Multigrid Methods*, John Wiley & Sons, New York.

Chapter 8

Abbott, M. B. and Basco, B. R. (1990). *Computational Fluid Dynamics – An Introduction for Engineers*, Longman Scientific & Technical, Harlow.

Ahmadi-Befrui, B., Gosman, A. D., Issa, R. I. and Watkins, A. P. (1990). EPISO – An Implicit Non-iterative Solution Procedure for the Calculation of Flows in Reciprocating Engine Chambers, *Comput. Methods Appl. Mech. Eng.*, Vol. 79, pp. 249–279.

Amsden, A. A. and Harlow, F. H. (1970). *The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows*, Los Alamos Scientific Laboratory Report LA-4370, Los Alamos, NM.

Amsden, A. A., Butler, T. D., O'Rourke, P. J. and Ramshaw, J. D. (1985). KIVA – A Comprehensive Model of 2D and 3D Engine Simulations, SAE Paper No. 850554.

Amsden, A. A., O'Rourke, P. J. and Butler, T. D. (1989). *KIVA-II – A Computer Program for Chemically Reactive Flows with Sprays*, Los Alamos National Laboratory Report LA-11560-MS.

Anderson, D. A., Tannehill, J. C. and Pletcher, R. H. (1984). *Computational Fluid Mechanics and Heat Transfer*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Blunsdon, C. A., Malalasekera, W. M. G. and Dent, J. C. (1992). Application of the Discrete Transfer Model of Thermal Radiation in CFD Simulation of Diesel Engine Combustion and Heat Transfer, SAE Paper No. 922305.

Blunsdon, C. A., Malalasekera, W. M. G. and Dent, J. C. (1993). Modelling Infrared Radiation from the Combustion Products in a SI Engine, SAE Paper No. 932699.

Crank, J. and Nicolson, P. (1947). A Practical Method for Numerical Evaluation of Solutions of Partial Differential Equations of the Heat-conduction Type, *Proc. Cambridge Phil. Soc.*, Vol. 43, pp. 50–67.

Fletcher, C. A. J. (1991). *Computational Techniques for Fluid Dynamics*, Vols I and II, Springer-Verlag, Berlin.
Harlow, F. H. and Amsden, A. A. (1971). A Numerical Fluid Dynamics Calculation Method for All Flow Speeds, *J. Comput. Phys.*, Vol. 8, pp. 197–213.
Harlow, F. H. and Welch, J. E. (1965). Numerical Calculation of Time-dependent Viscous Incompressible Flow of Fluid with Free Surface, *Phys. Fluids*, Vol. 8, pp. 2182–2189.
Hirt, C. W., Amsden, A. A. and Cook, J. L. (1974). An Arbitrary Lagrangian–Eulerian Computing Method for All Flow Speeds, *J. Comput. Phys.*, Vol. 14, pp. 227–253.
Issa, R. I. (1986). Solution of Implicitly Discretised Fluid Flow Equations by Operator-Splitting, *J. Comput. Phys.*, Vol. 62, pp. 40–65.
Issa, R. I., Gosman, A. D. and Watkins, A. P. (1986). The Computation of Compressible and Incompressible Recirculating Flows, *J. Comput. Phys.*, Vol. 62, pp. 66–82.
Kim, S. W. and Benson, T. J. (1992). Comparison of the SMAC, PISO and Iterative Time-advancing Schemes for Unsteady Flows, *Comput. Fluids*, Vol. 21, No. 3, pp. 435–454.
Özişik, M. N. (1985). *Heat Transfer – A Basic Approach*, McGraw-Hill, New York.
Zellat, M., Rolland, Th. and Poplow, F. (1990). Three-Dimensional Modelling of Combustion and Soot Formation in an Indirect Injection Diesel Engine, SAE Paper No. 900254.

Chapter 9

Jayatilleke, C. L. V. (1969). The Influence of Prandtl Number and Surface Roughness on the Resistance of the Laminar Sublayer to Momentum and Heat Transfer, *Prog. Heat Mass Transfer*, Vol. 1, p. 193.
Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.
Schlichting, H. (1979). *Boundary-layer Theory*, 7th edn, McGraw-Hill, New York.

Chapter 10

AIAA (1998). *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations*, AIAA Guide G-077-1998.
Chen, Q. and Srebric, J. (2001). *How to Verify, Validate and Report Indoor Environment Modelling CFD Analyses*, Final Report ASHRAE RP-1133, Welsh School of Architecture, Cardiff University, UK, and Dept. of Architectural Engineering, Pennsylvania State University, USA.
Chen, Q. and Srebric, J. (2002). A Procedure for Verification, Validation, and Reporting of Indoor Environment CFD Analyses, *Int. J. HVAC&R Res.*, Vol. 8, No. 2, pp. 201–216.
Coleman, H. W. and Stern, F. (1997). Uncertainties and CFD Code Validation, *J. Fluids Eng., Trans. ASME*, Vol. 199, pp. 795–803.
ERCOFTAC (2000). *Best Practice Guidelines, Version 1.0*, M. Casey and T. Wintergerste (eds), ERCOFTAC Special Interest Group on Quality and Trust Industrial CFD.
Oberkampf, W. L. and Trucano, T. G. (2002). Verification and Validation in Computational Fluid Dynamics, *Prog. Aerosp. Sci.*, Vol. 38, pp. 209–272.

Roache, P. (1997). Quantification of Uncertainty in Computational Fluid Dynamics, *Ann. Rev. Fluid Mech.*, Vol. 29, pp. 123–160.

Roache, P. (1998). *Verification and Validation of Computational Fluid Dynamics Simulations*, Hermosa, Albuquerque, NM.

Srebric, J. and Chen, Q. (2002). An Example of Verification, Validation, and Reporting of Indoor Environment CFD Analyses, *ASHREA Trans. Res.*, Paper 4569 (RP-1133), Vol. 108, No. 2, pp. 185–194.

Chapter 11

Amsden, A. A. (1997). *A Block Structured KIVA Program for Engines with Vertical or Canted Valves*, Los Alamos National Laboratory Report No. LA-UR-97-698.

Anderson, W. K. and Bonhaus, D. L. (1994). An Implicit Upwind Algorithm for the Computation of Turbulent Flows on Unstructured Grids, *Comput. Fluids*, Vol. 23, No. 1, pp. 1–21.

Bird, R. B., Stewart, W. E. and Lightfoot, E. N. (2002). *Transport Phenomena*, 2nd edn, John Wiley & Sons, New York.

Cabello, J., Morgan, K. and Lohner, R. A. (1994). Comparison of Higher-Order Schemes in a Finite Volume Solver for Unstructured Grids, AIAA Paper 94-2295, 25th Fluid Dynamics Conference, Colorado Springs, CO.

Courier, W. J. and Powell, K. G. (1996). Solution Adaptive Cartesian Cell Approach for Viscous and Invisid Flow, *AIAA J.*, Vol. 34, No. 5, pp. 938–945.

Darwish, M. S. and Moukalled, F. (2003). TVD Schemes for Unstructured Grids, *Int. J. Heat Mass Transfer*, Vol. 46, pp. 599–611.

Davidson, L. (1996). A Pressure Correction Method for Unstructured Meshes with Arbitrary Control Volumes, *Int. J. Numer. Methods Fluids*, Vol. 22, pp. 265–281.

Demirdzic, I. (1982). A Finite Volume Method for Computation of Fluid Flow in Complex Geometries, Ph.D. Thesis, Imperial College, London.

Demirdzic, I., Gosman, A. D., Issa, R. I. and Peric, M. (1987). A Calculation Procedure for Turbulent Flow in Complex Geometries, *Comput. Fluids*, Vol. 15, No. 3, pp. 251–273.

Ferziger, J. H. and Peric, M. (2001). *Computational Methods for Fluid Dynamics*, 3rd rev. edn, Springer-Verlag, New York.

Golub, G. H. and Van Loan, C. F. (1989). *Matrix Computation*, 2nd edn, Johns Hopkins University Press, Baltimore, MD.

Haselbacher, A. (1999). A Grid-transparent Numerical Method for Compressible Viscous Flows on Mixed Unstructured Grid, Ph.D. Thesis, Loughborough University.

Haselbacher, A. and Blazek, J. (2000). Accurate and Efficient Discretisation of Navier–Stokes Equations on Mixed Grids, *AIAA J.*, Vol. 38, No. 11, pp. 2094–2102.

Henson, J. C. (1998). Numerical Simulation of S.I. Engines with Special Emphasis on Radiative Heat Transfer, Ph.D. Thesis, Loughborough University.

Hirt, C. W., Amsden, A. A. and Cook, J. L. (1974). An Arbitrary Lagrangian-Eulerian Computing Method for All Flow Speeds, *J. Comput. Phys.*, Vol. 14, pp. 227–253.

Karki, K. C. and Patankar, S. V. (1988). Calculation Procedure for Viscous Incompressible Flows in Complex Geometries, *Numer. Heat Transfer*, Vol. 14, pp. 295–307.

Kim, D. and Choi, H. (2000). A Second-Order Time Accurate Finite Volume Method for Unsteady Incompressible Flow in Hybrid Unstructured Grids, *J. Comput. Phys.*, Vol. 162, pp. 411–428.

Kordula, W. and Vinokur, M. (1983). Efficient Computation of Volume in Flow Predictions, *AIAA J.*, Vol. 21, pp. 917–918.
Mathur, S. R. and Murthy, J. Y. (1997). A Pressure-Based Method for Unstructured Meshes, *Numer. Heat Transfer Part B*, Vol. 31, pp. 195–215.
Peric, M. (1985). Finite Volume Method for the Prediction of Three-dimensional Fluid Flow in Complex Ducts, Ph.D. Thesis, Imperial College, London.
Reggio, M. and Camarero, R. (1986). Numerical Solution Procedure for Viscous Incompressible Flows, *Numer. Heat Transfer*, Vol. 10, pp. 131–146.
Rhie, C. M. and Chow, W. L. (1983). Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation, *AIAA J.*, Vol. 21, No. 11, pp. 1525–1532.
Rodi, W., Majumdar, S. and Schonung, B. (1989). Finite Volume Methods for Two-dimensional Incompressible Flows with Complex Boundaries, *Comput. Methods Appl. Mech. Eng.*, Vol. 75, pp. 369–392.
Shyy, W. and Vu, T. G. (1991). On the Adaptation of Velocity Variable and Grid Systems for Fluid Flow in Curvilinear Co-ordinates, *J. Comput. Phys.*, Vol. 92, pp. 82–105.
Shyy, W., Correa, S. M. and Braaten, M. E. (1988). Computation of Flow in a Gas Turbine Combustor, *Combust. Sci. Technol.*, Vol. 58, No. 1–3, pp. 97–117.
Thomson, J. F. (1984). Grid Generation Techniques in Computational Fluid Dynamics, *AIAA J.*, Vol. 22, No. 11, pp. 1505–1523.
Thomson, J. F. (1988). Grid Generation, in W. J. Minkowycz, E. M. Sparrow, G. E. Schneider and R. H. Pletcher (eds) *Handbook of Numerical Heat Transfer*, John Wiley & Sons, New York, Chapter 21.
Whitaker, D. L., Grossman, B. and Löhner, R. (1989). Two-dimensional Euler Computation on a Triangular Mesh Using Upwind Finite-volume Scheme, AIAA Paper 89-0479, 27th Aerospace Science Meeting, Reno, NV.

Chapter 12

Abramowitz, M. and Stegun, I. A. (1970). *Handbook of Mathematical Functions*, Dover, New York.
Barlow, R. S. (2000). TNF Website of the International Workshop on Measurements and Computation of Turbulent Nonpremixed Flames, http://www.ca.sandia.gov/tdf/Workshop.html.
Barlow, R. S., Fiechtner, G. J., Carter, C. D. and Chen, J. Y. (2000). Instantaneous and Mean Compositional Structure of Bluff-body Stabilized Nonpremixed Flames, *Combust. Flame*, Vol. 120, pp. 549–569.
Barlow, R. S., Karpetis, A. N., Frant, J. H. and Chen, J. Y. (2001). Scalar Profiles and NO Formation in Laminar Opposed-flow Partially Premixed Methane/Air Flames, *Combust. Flame*, Vol. 127, pp. 2101–2118.
Bartok, W. and Sarofim, A. F. (eds) (1991). *Fossil Fuel Combustion: A Source Book*, Wiley Interscience, New York.
Baulch, D. L., Cobos, C. J., Cox, R. A., Frank, P., Hayman, G., Just, T. H., Kerr, J. A., Murrells, T., Pilling, M. J., Troe, J., Walker, R. W. and Warnantz, J. (1994). Summary Table of Evaluated Kinetic Data for Combustion Modelling: Supplement 1, *Combust. Flame*, Vol. 98, pp. 59–74.
Beck, J. C. and Watkins, A. P. (2004). The Simulation of Fuel Sprays Using the Moments of the Drop Number Size Distribution. *Int. J. Engine Res.*, Vol. 5, No. 1, pp. 1–21.
Beeri, Z., Blunsdon, C. A., Malalasekera, W. and Dent, J. C. (1996). Comprehensive Modelling of Turbulent Flames with the Coherent Flame-Sheet Model – Part II:

High Momentum Reactive Jets, *ASME J. Energy Resour. Technol.*, Vol. 118, pp. 72–76.

Bennett, B. A. and Smooke, M. D. (1998). Local Rectangular Refinement with Applications to Axisymmetric Laminar Flames, *Combust. Theory Modelling*, Vol. 2, pp. 221–258.

Bilger, R. W. (1976). Turbulent Jet Diffusion Flames, *Prog. Energ. Combust. Sci.*, Vol. 1, pp. 87–109.

Bilger, R. W. (1980). Turbulent Flows with Non-premixed Reactants, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Topics in Applied Physics, Springer-Verlag, Berlin, Chapter 3, pp. 65–114.

Bilger, R. W. (1988). The Structure of Turbulent Non-premixed Flames, *Twenty Second Symposium (International) on Combustion*, The Combustion Institute, pp. 475–488.

Bilger, R. W. (1993). Conditional Moment Closure for Turbulent Reacting Flow, *Phys. Fluids* A, Vol. 5, No. 2, pp. 436–444.

Bilger, R. W. (2000). Future Progress in Turbulent Combustion Research. *Prog. Energy Combust. Sci.*, Vol. 26, pp. 367–380.

Bilger, R. W. and Kent, J. H. (1974). Concentration Fluctuations in Turbulent Jet Flames, *Combust. Sci. Technol.*, Vol. 9, p. 25.

Blunsdon, C. A., Malalasekera, W. and Dent, J. C. (1992). Application of the Discrete Transfer Model of Thermal Radiation in a CFD Simulation of Diesel Engine Combustion and Heat Transfer, SAE Conference, International Fuel and Lubricants Meeting and Exposition, San Francisco, USA, SAE Paper No. 922305.

Blunsdon, C. A., Dent, J. C. and Malalasekera, W. (1993). Modelling Infrared Radiation from the Combustion Products in a Spark Ignition Engine, SAE Fuels & Lubricants Meeting and Exposition, Philadelphia, USA, SAE Paper No. 932699.

Blunsdon, C. A., Beeri, Z., Malalasekera, W. and Dent, J. C. (1996). Comprehensive Modelling of Turbulent Flames with the Coherent Flame-Sheet Model – Part 1: Buoyant Diffusion Flames, *ASME J. Energy Resour. Technol.*, Vol. 118, pp. 65–71.

Bowman, C. T., Hanson, R. K., Davidson, D. F., Gardiner, W. C. Jr, Lissianski, V., Smith, G. P., Golden, D. M., Frenklach, M. and Goldenberg, M. (1996). http://www.me.berkeley.edu/gri_mech/

Branley, N. and Jones, W. P. (2001). Large Eddy Simulation of a Turbulent Non-premixed Flame, *Combust. Flame*, Vol. 127, pp. 1914–1934.

Bray, K. N. C. and Peters, N. (1994). Laminar Flamelets in Turbulent Flames, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Academic Press, New York, Chapter 2.

Bray, K. N. C., Champion, M. and Libby, P. A. (1994). Flames in Stagnating Turbulence, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Academic Press, New York, Chapter 9.

Cengel, Y. A. and Boles, M. A. (2002). *Thermodynamics: An Engineering Approach*, 4th int. edn, McGraw-Hill Higher Education, New York.

Chase, M. W. Jr, Davies, C. A., Davies, J. R. Jr, Fulrip, D. J., McDonald, R. A. and Syverud, A. N. (1985). *JANAF Thermochemical Tables*, 3rd edn, *J. Phys. Chem. Ref. Data*, Vol. 14, Suppl. 1.

Chen, C.-S., Chang, K. C. and Chen, J.-Y. (1994). Application of a Robust β-pdf Treatment to Analysis of Thermal NO Formation in Nonpremixed Hydrogen-Air Flame, *Combust. Flame*, Vol. 98, pp. 375–390.

Chen, J.-Y. and Chang, W. C. (1996). Flamelet Modelling of CO and NO_x emission from a Turbulent, Methane Hydrogen Jet Nonpremixed flame, *Proc. Combust. Inst.*, Vol. 26, pp. 2207–2214.

Clarke, A. (2002). Calculation and Consideration of the Lewis Number for Explosion Studies, *Trans. IChemE*, Vol. 80, Pt B, pp. 135–140.

Conaire, M. O., Curran, H. J., Simmie, J. M., Pitz, W. J. and Westbrook, C. K. (2004). A Comprehensive Modelling Study of Hydrogen Oxidation, *Int. J. Chem. Kinet*., Vol. 36, pp. 603–622.

Cook, A. and Riley, J. J. (1998). Subgrid-scale Modelling of Turbulent Reacting Flows, *Combust. Flame*, Vol. 112, pp. 593–606.

Dally, B. B., Fletcher, D. F. and Masri, A. R. (1998a). Flow and Mixing Fields of Turbulent Bluff Body Jets and Flames. *Combust. Theory Modelling*, Vol. 2, pp. 193–219.

Dally, B. B., Masri, A. R., Barlow, R. W. and Fiechtner, G. J. (1998b). Instantaneous and Mean Compositional Structure of Bluff-body Stabilized Nonpremixed Flames. *Combust. Flame*, Vol. 114, pp. 119–148.

Dally, B. B., Masri, A. R., Barlow, R. W., Fiechtner, G. J. and Fletcher, D. F. (1996). Measurements of NO in Turbulent Non-premixed Flames Stabilized on a Bluff Body. *Proc. Combust. Inst*., Vol. 26, pp. 2191–2197.

DesJardin, P. E. and Frankel, S. H. (1998). Large Eddy Simulation of a Nonpremixed Reacting Jet: Application and Assessment of Subgrid-Scale Combustion Models, *Phys. Fluids*, Vol. 10, No. 9, pp. 2298–2314.

Dixon-Lewis, G., David, T., Gaskell, P. H., Fukutani, S., Jinno, H., Miller, J. A., Kee, R. J., Smooke, M. D., Peters, N., Effelsberg, E., Warnatz, J. and Behrendt, F. (1984). Calculation of the Structure and Extinction of Limit of a Methane-Air Counterflow Diffusion Flame in the Forward Stagnation Region of a Porous Cylinder, *Twentieth Symposium (International) on Combustion*, The Combustion Institute, pp. 1893–1904.

Dopazo, C. (1994). Recent Developments in pdf Methods, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Academic Press, New York, Chapter 7.

Drake, M. and Blint, R. J. (1989). Thermal NO_x in Stretched Laminar Opposed-Flow Diffusion Flames with $CO/H_2/N_2$ Fuel, *Combust. Flame*, Vol. 76, pp. 151–167.

Drake, M. C. and Blint, R. J. (1988). Structure of Laminar Opposed-flow Diffusion Flames with $CO/H_2/N_2$ Fuel, *Combust. Sci. Technol.*, Vol. 61, pp. 187–224.

Dryer, F. L. (1991). The Phenomenology of Modelling Combustion Chemistry, in W. Bartok and A. F. Sarofim (eds) *Fossil Fuel Combustion: A Source Book*, Wiley Interscience, New York, Chapter 3.

Dupont, V., Pourkashanian, M., Richardson, A. P., Williams, A. and Scott, M. J. (1995). The Importance of Prompt-NO Formation and of NO Reconversion in Strained Binary Rich Partially Premixed Flames, *18th International Symposium on Transport Phenomena in Combustion*, San Francisco, USA, pp. 263–274.

Effelsberg, E. and Peters, N. (1988). Scalar Dissipation Rates in Turbulent Jets and Jet Diffusion Flames, *Twenty-Second Symposium (International) on Combustion*, The Combustion Institute, pp. 693–700.

Eickhoff, H. and Grethe, K. (1979). A Flame-zone Model for Turbulent Hydrocarbon Diffusion Flames, *Combust. Flame*, Vol. 35, pp. 267–275.

Ertesvag, I. S. and Magnussen, B. F. (2000). The Eddy Dissipation Turbulence Energy Cascade Model, *Combust. Sci. Technol.*, Vol. 159, pp. 213–236.

Evans, M., Hastings, N. and Peacock, B. (2000). *Statistical Distributions*, 3rd edn, Wiley Series in Probability and Statistics, John Wiley & Sons, New York.

Favre, A. (1969). Statistical Equations of Turbulent Gases, in *Problems of Hydrodynamics and Continuum Mechanics*, SIAM, Philadelphia, pp. 231–266.

Fenimore, C. P. (1970). Formation of Nitric Oxide in Premixed Hydrocarbon Flames, *Thirteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 373–380.

Frenklach, M., Bowman, T., Smith, G. and Gardiner, W. (2004). GRI 3.0, http://www.me.berkeley.edu/gri_mech/version30/text30.html and Gas Research Institute, Chicago, USA.

Fukutani, S., Kunioshi, N. and Jinno, H. (1990). Flame Structure of an Axisymmetric Hydrogen-Air Diffusion Flame, *Twenty-Third (International) Symposium on Combustion*, The Combustion Institute, pp. 567–573.

Gardiner, W. C. Jr (ed.) (1984). *Combustion Chemistry*, Springer-Verlag, New York.

Gordon, S. and McBride, B. J. (1994). *Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications*, NASA Reference Publication 1311, NASA, USA.

Gosman, A. D., Lockwood, F. C. and Salooja, A. P. (1978). The Prediction of Cylindrical Furnaces Gaseous Fuelled with Premixed and Diffusion Burners, *Seventeenth Symposium (International) on Combustion*, The Combustion Institute, pp. 747–760.

Henson, J. C. and Malalasekera, W. (2000). Full-cycle Firing Simulation of a Pent-roof Spark-ignition Engine with Visualization of the Flow Structure, Flame Propagation and Radiative Heat Flux, *Proc. IMechE Part D: J. Engines*, Vol. 214, pp. 957–971.

Henson, R. K. and Salimian, S. (1984). Survey of Rate Constants in the N/H/O System, in W. C. Gardiner Jr (ed.) *Combustion Chemistry*, Springer-Verlag, New York, Chapter 6.

Hewson, J. C. and Bollig, M. (1996). Reduced Mechanisms for NO_x Emissions from Hydrocarbon Diffusion Flames, *Twenty-Sixth Symposium (International) on Combustion*, The Combustion Institute, pp. 2171–2180.

Hossain, M. (1999). CFD Modelling of Turbulent Nonpremixed Combustion, Ph.D. Thesis, Loughborough University.

Hossain, M. and Malalasekera, W. (2003). Modelling of a Bluff-body Stabilized CH_4/H_2 Flame based on Laminar Flamelet Model with Emphasis on NO Prediction, *Proc. IMechE, Part A: J. Power Energy*, Vol. 217, pp. 201–210.

Hossain, M., Jones, J. C. and Malalasekera, W. (2001). Modelling of a Bluff-body Nonpremixed Flame using a Coupled Radiation/Flamelet Combustion Model, *Flow, Turbul. Combust.*, Vol. 67, pp. 217–234.

Jones, W. P. and Priddin, C. H. (1978). Prediction of the Flow Field and Local Gas Composition in Gas Turbine Combustors, *Seventeenth Symposium (International) on Combustion*, The Combustion Institute, pp. 399–407.

Jones, W. P. and Whitelaw, J. H. (1982). Calculation Methods for Reacting Turbulent Flows: A Review, *Combust. Flame*, Vol. 48, No. 1, pp. 1–26.

Kee, R. J., Rupley, F. M., Meeks, J. A. and Miller, E. (1996). *Chemkin III: A Fortran Chemical Kinetics Package for the Analysis of Gas-phase Chemical and Plasma Kinetics*, Sandia National Laboratories Report, UC-405, SAND96, http://www.ca.sandia.gov/chemkin/index.html.

Kent, J. H. and Bilger, R. W. (1973). Turbulent Diffusion Flames, *Fourteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 615–625.

Kim, S. H. and Huh, K. Y. (2002). Use of Conditional Moment Closure Model to Predict NO Formation in a Turbulent CH_4/H_2 Flame over a Bluff-Body, *Combust. Flame*, Vol. 130, pp. 94–111.

Klimenko, A. Y. and Bilger, R. W. (1999). Conditional Moment Closure for Turbulent Combustion, *Prog. Energy Combust. Sci.*, Vol. 25, pp. 595–687.

Kuo, K. (1986). *Principles of Combustion*, John Wiley & Sons, New York.

Kuo, K. (2005). *Principles of Combustion*, 2nd edn, John Wiley & Sons, New York.

Lentini, D. (1994). Assessment of Stretched Laminar Flamelet Approach for Non-premixed Turbulent Combustion, *Combust. Sci. Technol.*, Vol. 100, pp. 95–122.

Lewis, M. H. and Smoot, L. D. (1981). Turbulent Gaseous Combustion. Part I: Local Species Concentration Measurements, *Combust. Flame*, Vol. 42, p. 183.

Libby, P. A. and Williams, F. A. (1994). Fundamental Aspects and a Review, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Academic Press, New York, Chapter 1.

Liu, F., Guo, H., Smallwood, G. J., Gölder, O. L. and Matovic, M. D. (2002). A Robust and Accurate Algorithm of the β-pdf Integration and Its Application to Turbulent Methane-Air Diffusion Combustion in a Gas Turbine Combustor Simulator, *Int. J. Thermal Sci.*, Vol. 41, pp. 763–772.

Lookwood, F. C. (1977). The Modelling of Turbulent Premixed and Diffusion Combustion in the Computation of Engineering Flows, *Combust. Flame*, Vol. 29, pp. 111–122.

Lockwood, F. C. and Monib, H. A. (1980). Fluctuating Temperature Measurements in a Heated Round Free Jet, *Combust. Sci. Technol.*, Vol. 22, pp. 63–81.

Lockwood, F. C. and Naguib, A. S. (1975). The Prediction of Fluctuations in the Properties of Free, Round Jet Turbulent Diffusion Flames, *Combust. Flame*, Vol. 24, pp. 109–124.

Lockwood, F. C. and Odidi, A. O. (1975). Measurement of Mean and Fluctuating Temperature and Ion Concentration in Round Jet Turbulent Diffusion and Premixed Flames, *Fifteenth Symposium (International) on Combustion*, The Combustion Institute, p. 561.

Lockwood, F. C., Salooja, A. P. and Syed, S. A. (1980). A Prediction Method for Coal-fired Furnaces, *Combust. Flame*, Vol. 38, pp. 1–15.

Lockwood, F. C., Rizvi, S. M. A. and Shah, N. G. (1986). Comparative Predictive Experience of Coal Firing, *Proc. IMechE*, Pt A, Vol. 100, pp. 79–83.

Lutz, A., Kee, R. J., Grcar, J. F. and Rupley, F. M. (1997). *OPPDIFF: A Fortran Program for Computing Opposed-flow Diffusion Flames*, Sandia Report SAND96-8243, Livermore.

Magnussen, B. F. (2005). The Eddy Dissipation Concept: Bridge Between Science and Technology, Invited Lecture, *Proceedings of the ECCOMAS Thematic Conference on Computational Combustion*, Lisbon.

Magnussen, B. F. and Hjertager, B. H. (1976). On the Mathematical Modelling of Turbulent Combustion with Special Emphasis on Soot Formation and Combustion, *Sixteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 719–729.

Marble, F. E. and Broadwell, J. E. (1977). *The Coherent Flame Sheet Model for Nonpremixed Turbulent Combustion*, Project SQUID, Report No. TRW-9-PU, Purdue University.

Masri, A. R. (1996). Database for Bluff Body Flames, Department of Mechanical and Mechatronics Engineering, The University of Sydney, http://www.mech.eng.usyd.edu.au/research/energy/resources.html.

Masri, A. R., Dibble, R. W. and Barlow, R. S. (1996). The Structure of Turbulent Nonpremixed Flames Revealed by Raman-Rayleigh-LIF Measurements, *Prog. Energy Combust. Sci.*, Vol. 22, pp. 307–362.

Massias, A., Diamantis, D., Mastorakos, E. and Goussis, D. A. (1999). Global Reduced Mechanisms for Methane and Hydrogen Combustion with Nitric Oxide Formation Constructed with CSP Data, *Combust. Theory Modelling*, Vol. 3, pp. 233–257.

Mauss, F., Keller, D. and Peters, N. (1990). A Lagrangian Simulation of Flamelet Extinction and Re-ignition in Turbulent Jet Diffusion Flames, *Twenty-Third Symposium (Int.) on Combustion*, The Combustion Institute, pp. 693–698.

McBride, B. J. (2004). CEA Website, http://www.grc.nasa.gov/WWW/CEAWeb/

McGuirk, J. J. and Rodi, W. (1979). The Calculation of Three-dimensional Free Jets, in F. Durst, B. E. Launder, F. W. Schmidt and J. H. Whitelaw (eds) *1st Symposium on Turbulent Shear Flows*, Springer-Verlag, New York, pp. 71–83.

Miller, J. A. and Bowman, C. T. (1989). Mechanism and Modelling of Nitrogen Chemistry in Combustion, *Prog. Energy Combust. Sci.*, Vol. 15, pp. 287–338.

Morley, C. (2005). GASEQ Windows-based Computer Program for Equilibrium Calculations, http://www.gaseq.co.uk/

Murthy, R. V. V. S., Malalasekera, W. and Hossain, M. (2006). A Laminar Flamelet Based NO_x-Radiation Integrated Modelling of Turbulent Non-Premixed Flame, *Turbulence, Heat and Mass Transfer 5, Proceedings of the Fifth International Symposium on Turbulence, Heat and Mass Transfer*, K. Hanjalic, Y. Nagano and S. Jakirlic (eds), Dubrovnik, Croatia, 25–29 September, 2006, Begell-House Inc., New York, pp. 609–612. Also available in CD-ROM.

Nazha, M. A. A., Rajakaruna, H. and Malalasekera, W. (2001). Effects of Radiation on Predicted Flame Temperature and Combustion Products of a Burning Liquid Fuel Spray, *Sixth International Conference on Combustion Technologies for a Clean Environment*, Oporto, Vol. II, Paper No. 20.1, pp. 639–643.

Nickjooy, M., So, R. M. C. and Peck, R. E. (1988). Modelling of Jet-swirl-stabilised Reacting Flows in Axisymmetric Combustors, *Combust. Sci. Technol.*, Vol. 25, No. 1, pp. 63–75.

Paul, P. and Warnatz, J. (1998). A Re-evaluation of the Means Used to Calculate Transport Properties of Reacting Flows, *Twenty-seventh (International) Symposium on Combustion*, The Combustion Institute, pp. 495–504.

Peters, N. (1984). Laminar Flamelet Model in Non-premixed Turbulent Combustion. *Prog. Energy Combust. Sci.*, Vol. 10, pp. 319–339.

Peters, N. (1986). Laminar Flamelet Concepts in Turbulent Combustion. *Proc. Combust. Inst.*, Vol. 21, pp. 1231–1250.

Peters, N. (1991). Length Scales in Laminar and Turbulent Flames, in E. S. Oran and J. P. Boris (eds) *Numerical Approaches to Combustion Modelling*, *Prog. Astronaut. Aeronaut.*, AIAA, pp. 155–182.

Peters, N. (1993). Flame Calculation with Reduced Mechanisms, in N. Peters and B. Rogg (eds) *Reduced Kinetic Mechanisms for Applications in Combustion Systems*, Springer-Verlag, Berlin, Chapter 1.

Peters, N. (2000). *Turbulent Combustion*, Cambridge University Press, Cambridge, Chapter 3.

Pitsch, H. (1998). FlameMaster: A C++ Program for 0D Combustion and 1D Laminar Flame Calculations, http://www.stanford.edu/~hpitsch.

Pitsch, H. and Peters, N. (1998). A Consistent Flamelet Formulation for Nonpremixed Combustion Considering Differential Diffusion Effects, *Combust. Flame*, Vol. 114, pp. 26–40.

Poinsot, T. and Veynante, D. (2005). *Theoretical and Numerical Combustion*, 2nd edn, Edwards, Philadelphia, PA.

Pope, S. B. (1976). The Probability Approach to Modelling of Turbulent Reacting Flows, *Combust. Flame*, Vol. 27, pp. 299–312.

Pope, S. B. (1978). An Explanation of the Turbulent Round-jet/Plane-jet Anomaly. *AIAA J.*, Vol. 16, pp. 279–281.

Pope, S. B. (1985). PDF Methods for Turbulent Reacting Flows, *Prog. Energy Combust. Sci.*, Vol. 11, pp. 119–192.

Pope, S. B. (1990). Computation of Turbulent Combustion: Progress and Challenges, *Twenty-Third Symposium (International) on Combustion*, The Combustion Institute, pp. 591–612.

Pope, S. B. (1991). Combustion Modelling using Probability Density Function Methods, in E. S. Oran and J. P. Boris (eds) *Numerical Approaches to Combustion Modelling*, *Prog. Astronaut. Aeronaut*, AIAA, Chapter 11.

Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T. (1993). *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge University Press, Cambridge.

Radhakrishnan, K. and Pratt, D. T. (1988). Fast Algorithm for Calculating Chemical Kinetics in Turbulent Reacting Flow, *Combust. Sci. Technol.*, Vol. 58, pp. 155–176.

Reid, R. C., Prausnitz, J. M. and Poling, B. E. (1987). *The Properties of Gases and Liquids*, 4th edn, McGraw-Hill, New York.

Rogers, G. F. C. and Mayhew, Y. R. (1994). *Thermodynamic and Transport Properties of Fluids*, 5th edn, Basil Blackwell, Oxford.

Rogg, B. (1993). RUN-1DL: The Cambridge Universal Laminar Flamelet Code, in N. Peters and B. Rogg (eds) *Reduced Kinetic Mechanisms for Applications in Combustion Systems*, Springer Verlag, Berlin, Appendix C.

Rogg, B. and Wang, W. (1997). *RUN-1DL: The Laminar Flame and Flamelet Computer Code. User Manual*, Lehrsthul für Strömungsmechanik, Institut für Thermo- und Fluiddynamik, Ruhr-Universität Bochum, Bochum.

Sanders, J. P. H. and Gökalp, I. G. (1995). Flamelet Based Predictions and Scaling Laws of NO Formation in Turbulent Hydrogen Diffusion Flames, *Eighth International Symposium on Transport Phenomena in Combustion*, San Francisco, pp. 286–297.

Selle, L., Lartigue, G., Poinsot, T., Koch, R., Schildmacher, K.-U., Krebs, W., Prade, B., Kaufmann, P. and Veynante, D. (2004). Compressible Large Eddy Simulation of Turbulent Combustion in Complex Geometry on Unstructured Meshes, *Combust. Flame*, Vol. 137, pp. 489–505.

Seshadri, K. and Williams, F. A. (1994). Reduced Chemical Systems and Their Application in Turbulent Combustion, in P. A. Libby and F. A. Williams (eds) *Turbulent Reacting Flows*, Academic Press, New York, Chapter 4.

Seshadri, K., Mauss, F., Peters, N. and Warnatz, J. (1990). A Flamelet Calculation of Benzene Formation in Co-flowing Laminar Diffusion Flames, *Twenty-Third Symposium (International) on Combustion*, The Combustion Institute, pp. 559–566.

Sick, V., Arnold, A., Dießel, E., Dreirer, T., Ketterle, W., Lange, B., Wolfrum, J., Thiele, K. U., Behrendt, F. and Warnatz, J. (1991). Two-dimensional Laser Diagnostic and Modelling of Counterflow Diffusion Flames, *Twenty-Third Symposium (International) on Combustion*, The Combustion Institute, p. 495.

Smith, G. P., Golden, D. M., Frenklach, M., Moriarty, N. W., Eiteneer, B., Goldenberg, M., Bowman, C. T., Hanson, R. K., Song, S., Gardiner Jr, W. C. Lissianski, V. V. and Qin, Z. (2003). GRI-Mech 3.0., http://www.me.berkeley.edu/gri-mech/

Smith, N. S. A., Bilger, R. W. and Chen, J.-Y. (1992). Modelling of Nonpremixed Hydrogen Jet Flames Using a Conditional Moment Closure Method, *Twenty-Fourth Symposium (International) on Combustion*, The Combustion Institute, pp. 263–269.

Smooke, M. D. (1991). Numerical Modelling of Laminar Diffusion Flames, in E. S. Oran and J. P. Boris (eds) *Numerical Approaches to Combustion Modelling*, *Prog. Astronaut. Aeronaut.*, AIAA, Chapter 7.

Smooke, M. D. and Bennett, B. A. (2001). Numerical Modelling of Multi-dimensional Laminar Flames, in C. E. Baukal, V. Y. Gershtein and X. Li (eds) *Computational Fluid Dynamics in Industrial Combustion*, CRC Press, Boca Raton, FL, Chapter 6.

Smooke, M. D., Puri, I. K. and Seshadri, K. (1986). A Comparison Between Numerical Calculation and Experimental Measurements of the Structures of a Counterflow Diffusion Flame Burning Diluted Methane in Diluted Air, *Twenty-first Symposium (International) on Combustion*, The Combustion Institute, pp. 1783–1792.

Smooke, M. D., Lin, P., Lam, J. K. and Long, M. B. (1990). Computational and Experimental Study of a Laminar Axisymmetric Methane-air Diffusion Flame, *Twenty-Third Symposium (International) on Combustion*, The Combustion Institute, pp. 575–582.

Spalding, D. B. (1971). Mixing and Chemical Reaction in Steady Confined Turbulent Flames, *Thirteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 649–657.

Spalding, D. B. (1979). *Combustion and Mass Transfer*, Pergamon Press, Oxford.

Swaminathan, N. and Bilger, R. W. (1999). Assessment of Combustion Submodels for Turbulent Nonpremixed Hydrocarbon Flames, *Combust. Flame*, Vol. 116, pp. 519–545.

Tsuji, H. and Yamaoka, I. (1967). The Counterflow Diffusion Flame in the Forward Stagnation Region of a Porous Cylinder, *Eleventh Symposium (International) on Combustion*, The Combustion Institute, pp. 979–984.

Tsuji, H. and Yamaoka, I. (1971). Structure Analysis of Counterflow Diffusion Flames in the Forward Stagnation Region of a Porous Cylinder, *Thirteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 723–731.

Turns, S. R. (1995). Understanding NO_x Formation in Nonpremixed Flames: Experiments and Modelling. *Prog. Energy Combust. Sci.*, Vol. 21, pp. 361–385.

Turns, S. R. (2000). *An Introduction to Combustion: Concepts and Applications*, 2nd edn, McGraw-Hill, New York.

Veynante, D. and Vervisch, L. (2002). Turbulent Combustion Modelling, *Prog. Energy Combust. Sci.*, Vol. 28, pp. 193–266.

Vranos, A., Knight, B. A., Proscia, W. M., Chiappetta, L. and Smooke, M. D. (1992). Nitric Oxide Formation and Differential Diffusion in a Methane Hydrogen Diffusion Flame, *Proc. Combust. Inst.*, Vol. 24, pp. 377–384.

Warnatz, J., Maas, U. and Dibble, R. W. (1996). *Combustion: Physical and Chemical Fundamentals, Modelling and Simulation, Experiment, Pollutant Formation*, Springer-Verlag, Berlin.

Warnatz, J., Maas, U. and Dibble, R. W. (2001). *Combustion: Physical and Chemical Fundamentals, Modelling and Simulation, Experiments, Pollution Formation*, 3rd edn, Springer-Verlag, Berlin.

Williams, F. A. (1985). *Combustion Theory*, 2nd edn, Addison-Wesley, Redwood City, CA.

Williams, F. A. (2000). Progress in Knowledge of Flamelet Structure and Extinction, *Prog. Energy Combust. Sci.*, Vol. 26, pp. 657–68.

Chapter 13

Baek, S. W. and Kim, M. Y. (1998). Radiative Heat Transfer in a Body-Fitted Axisymmetric Cylindrical Enclosure, *J. Thermophys. Heat Transfer*, Vol. 12, No. 4, pp. 596–599.

Baek, S. W., Kim, M. Y. and Kim, J. S. (1998). Nonorthogonal Finite-volume Solutions of Radiative Heat Transfer in a Three-Dimensional Enclosure, *Numer. Heat Transfer*, Pt B, Vol. 34, No. 4, pp. 419–437.

Brewster, M. Q. (1992). *Thermal Radiative Heat Transfer and Properties*, Wiley-Interscience, New York.

Carvalho, M. G. and Farias, T. L. (1998). Modelling of Heat Transfer in Radiating and Combustion Systems, *J. Chem. Eng. Res. Des., Trans. IChemE*, Pt A, Vol. 76, pp. 175–184.

Carvalho, M. G., Farias, T. and Fontes, P. (1991). Predicting Radiative Heat Transfer in Absorbing, Emitting, and Scattering Media using the Discrete Transfer Method, *ASME FED*, Vol. 160, pp. 17–26.

Chai, J. C. and Modar, J. P. (1996). Spatial-multiblock Procedure for Radiation Heat Transfer, *ASME HTD*, Vol. 332, pp. 119–127.

Chai, J. C., Lee, H. S. and Patankar, S. V. (1994a). Finite-volume Method for Radiation Heat Transfer, *J. Thermophys. Heat Transfer*, Vol. 8, No. 3, pp. 419–425.

Chai, J. C., Parthasarathy, G., Lee, H. S. and Patankar, S. V. (1994b). Finite-volume Radiation Heat Transfer Procedure for Irregular Geometries, *J. Thermophys. Heat Transfer*, Vol. 9, No. 3, pp. 410–415.

Chandrasekhar, S. (1960). *Radiative Transfer*, Dover, New York.

Charette, A., Sakami, M. and Le Dez, V. (1997). Analysis of Radiative Heat Transfer in Enclosures of Complex Geometry Using the Discrete Ordinates Method, Radiative Heat Transfer-II, in M. P. Mengüç (ed.) *Proceedings of the Second International Symposium on Radiation Transfer*, Kusadasi, Turkey, Begell House, Redding, CT, pp. 253–270.

Chui, E. H. and Raithby, G. D. (1993). Computation of Radiative Heat Transfer on a Non-orthogonal Mesh Using the Finite Volume Method, *Numer. Heat Transfer*, Pt B, Vol. 23, pp. 269–288.

Chui, E. H., Hughes, P. M. and Raithby, G. D. (1993). Implementation of the Finite Volume Method for Calculating Radiative Transfer in a Pulverised Fuel Flame, *Combust. Sci. Techn.*, Vol. 92, pp. 225–242.

Chui, E. H., Raithby, G. D. and Hughes, P. M. (1992). Prediction of Radiative Transfer in Cylindrical Enclosures with the Finite Volume Method, *J. Thermophys. Heat Transfer*, Vol. 6, No. 4, pp. 605–611.

Coelho, P. J. and Carvalho, M. G. (1997). A Conservative Formulation of the Discrete Transfer Method, *J. Heat Transfer*, Vol. 115, pp. 486–489.

Cumber, P. S. (1995). Improvements to the Discrete Transfer Method of Calculating Radiative Heat Transfer, *Int. J. Heat Mass Transfer*, Vol. 38, No. 3, pp. 2251–2258.

Denison, M. K. and Webb, B. W. (1993). A Spectral Line Based Weighted-sum-of-grey-gases Model Arbitrary RTE Solvers, *J. Heat Transfer*, Vol. 115, pp. 1004–1012.

Denison, M. K. and Webb, B. W. (1995). The Spectral Line Based Weighted-sum-of-grey-gases Model in Non-isothermal Non-homogeneous Media, *J. Heat Transfer*, Vol. 177, pp. 359–365.

Farmer, J. T. (1995). Improved Algorithms for Monte Carlo Analysis of Radiative Heat Transfer in Complex Participating Media, Ph.D. Thesis, The University of Texas at Austin.

Fiveland, W. A. (1982). A Discrete Ordinates Method for Predicting Radiative Heat Transfer in Axisymmetric Enclosures, ASME Paper 82-HT-20.

Fiveland, W. A. (1988). Three-dimensional Radiative Heat Transfer Solutions by the Discrete-ordinates Method, *J. Thermophys. Heat Transfer*, Vol. 2, No. 4, pp. 309–316.

Fiveland, W. A. (1991). The Selection of Discrete Ordinates Quadrature Sets for Anisotropic Scattering, *ASME HTD*, Vol. 160, pp. 89–96.

Fiveland, W. A. and Jessee, J. P. (1994). Comparisons of Discrete Ordinate Formulations for Radiative Heat Transfer in Multidimensional Geometries, *Radiative Heat Transfer: Current Research*, *ASME HTD*, Vol. 276, pp. 49–57.

Groshandler, W. L. (1993). RADCAL: A Narrow-Band Model for Radiation Calculation in a Combustion Environment, NIST Technical Note 1402, Fire Science Division, National Institute of Standards and Technology (NIST), Gaithersburg, MD, http://fire.nist.gov/bfrlpubs/

Henson J. C. (1998). Numerical Simulation of SI Engines with Special Emphasis on Radiative Heat Transfer, Ph.D. Thesis, Loughborough University.

Henson, J. C. and Malalasekera, W. (1997a). Comparison of Discrete Transfer and Monte Carlo Methods for Radiative Heat Transfer in Three-dimensional Non-homogeneous Scattering Media, *Numer. Heat Transfer*, Pt A, Vol. 32, No. 1, pp. 19–36.

Henson, J. C. and Malalasekera, W. (1997b). Benchmark Comparisons with Discrete Transfer Method Solutions for Radiative Heat Transfer in Three-dimensional, Nongrey, Scattering Media, in M. P. Mengüç (ed.) *Radiative Heat Transfer – II, Proceedings of the Second International Symposium on Radiation Transfer*, Kusadasi, Turkey, Begell House, Redding, CT, pp. 195–207.

Howell, J. R. (1988). Thermal Radiation in Participating Media: The Past, the Present and Some Possible Futures, *J. Heat Transfer*, Vol. 110, pp. 1220–1226.

Howell, J. R. (1998). The Monte Carlo Method in Radiative Heat Transfer, *J. Heat Transfer*, Vol. 120, pp. 547–560.

Howell, J. R. and Perlmutter, M. (1964). Monte Carlo Solution of Thermal Transfer in Nongrey Non-isothermal Gas with Temperature Dependent Properties, *AIChE J.*, Vol. 10, No. 4, pp. 562–567.

Hsu, P. F. and Tan, Z. (1997). Radiative and Combined-mode Heat Transfer within L-shaped Nonhomogeneous and Nongrey Participating Media, *Numer. Heat Transfer*, Pt A, Vol. 31, pp. 819–835.

Hsu, P. F., Tan, Z. M. and Howell, J. R. (1993). Radiative Transfer by the YIX Method in Nonhomogeneous Participating Media, *J. Thermophys. Heat Transfer*, Vol. 7, No. 3, pp. 487–495.

Hyde, D. J. and Truelove, J. S. (1977). *The Discrete Ordinates Approximation for Multidimensional Radiant Heat Transfer in Furnaces*, Technical Report, UKAEA No. AERE-R-8502, AERE Harwell.

Jamaluddin, A. S. and Smith, P. J. (1988). Predicting Radiative Transfer in Rectangular Enclosures Using the Discrete Ordinates Method, *Combust. Sci. Technol.*, Vol. 59, pp. 321–340.

Lathrop, K. D. and Carlson, B. G. (1965). *Discrete-ordinates Angular Quadrature of the Neutron Transport Equation*, Technical Information Series Report LASL-3186, Los Alamos National Laboratory.

Lockwood, F. C. and Shah, N. G. (1981). A New Radiation Solution Method for Incorporation in General Combustion Prediction Procedures, *Eighteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 1405–1414.

Mahan, J. R. (2002). *Radiation Heat Transfer: A Statistical Approach*, John Wiley & Sons, New York.

Malalasekera, W. and James, E. H. (1995). Calculation of Radiative Heat Transfer in Three-dimensional Complex Geometries, *ASME HTD*, Vol. 315, pp. 53–61.

Malalasekera, W. and James, E. H. (1996). Radiative Heat Transfer Calculations in Three-dimensional Complex Geometries, *J. Heat Transfer*, Vol. 118, pp. 225–228.

Malalasekera, W., Versteeg, H. K., Henson, J. C. and Jones, J. C. (2002). Calculation of Radiative Heat Transfer in Combustion Systems, *Clean Air*, Vol. 3, pp. 113–143.

Maltby, J. D. (1996). Coupled Monte Carlo/Finite Element Solution of Radiation – Conduction Problems in a Shaded Geometry, Paper Presented at the Open Forum Session on Radiation, ASME 31st National Heat Transfer Conference, Houston, Texas.

Maruyama, S. and Guo, Z. X. (2000). Radiative Heat Transfer in Homogeneous, Nongray and Ansiotropically Scattering Media, *Int. J. Heat Mass Transfer*, Vol. 43, No. 13, pp. 2325–2336.

Modest, M. F. (2003). *Radiative Heat Transfer*, Academic Press, London.

Murthy, J. Y. and Mathur, S. R. (1998). Finite Volume Method for Radiative Heat Transfer Using Unstructured Meshes, *J. Thermophys. Heat Transfer*, Vol. 12, No. 3, pp. 313–321.

Raithby, G. D. and Chui, E. H. (1990). A Finite Volume Method for Predicting Radiative Heat Transfer in Enclosures with Participating Media, *J. Heat Transfer*, Vol. 112, pp. 415–423.

Sakami, M., Charette, A. and Le Dez, V. (1996). Application of the Discrete Ordinates Method to Combined Conductive and Radiative Heat Transfer in a Two-Dimensional Complex Geometry, *J. Quant. Spectrosc. Radiat. Transfer*, Vol. 56, No. 4, pp. 517–533.

Sakami, M., Charette, A. and Le Dez, V. (1998). Radiative Heat Transfer in Three-dimensional Enclosures of Complex Geometry Using the Discrete Ordinates Method, *J. Quant. Spectrosc. Radiat. Transfer*, Vol. 59, Nos. 1–2, pp. 117–136.

Sarofim, A. F. (1986). Radiative Heat Transfer in Combustion: Friend or Foe, *Twenty-first Symposium (International) on Combustion*, The Combustion Institute, pp. 1–23.

Shah, N. G. (1979). New Method of Computation of Radiation Heat Transfer in Combustion Chambers, Ph.D. Thesis, Imperial College of Science and Technology, University of London.

Siegel, R. and Howell, J. R. (2002). *Thermal Radiation Heat Transfer*, 4th edn, Taylor & Francis, New York.

Versteeg, H. K., Henson, J. C. and Malalasekera, W. (1999a). Approximation Errors in the Heat Flux Integral of the Discrete Transfer Method Part 1: Transparent Media, *Numer. Heat Transfer*, Pt B, Vol. 36, pp. 387–407.

Versteeg, H. K., Henson, J. C. and Malalasekera, W. (1999b). Approximation Errors in the Heat Flux Integral of the Discrete Transfer Method Part 2: Absorbing/emitting Media, *Numer. Heat Transfer*, Pt B, Vol. 36, pp. 409–432.

Viskanta, R. and Mengüç, M. P. (1987). Radiation Heat Transfer in Combustion Systems, *Prog. Energy Combust. Sci.*, Vol. 2, pp. 97–160.

Appendix A

Abbott, M. B. and Basco, D. R. (1989). *Computational Fluid Dynamics – An Introduction for Engineers*, Longman Scientific & Technical, Harlow.

Fletcher, C. A. J. (1991). *Computational Techniques for Fluid Dynamics*, Vols I and II, Springer-Verlag, Berlin.

Appendix B

Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporation, Taylor & Francis Group, New York.

Appendix G

Abramowitz, M. and Stegun, A. (eds) (1964). *Handbook of Mathematical Functions*, Dover, New York.

Alvarez, N. J., Foote, K. L. and Pagni, P. J. (1984). Forced Ventilated Enclosure Fires, *Combust. Sci. Technol.*, Vol. 39. p. 55.

Benelli, G., Michele, G. D. E., Cossalter, V., Lio, M. D. A. and Rossi, G. (1992). Simulation of Large Non-linear Thermo-acoustic Vibration in a Pulsating Combustor, *Twenty-Fourth Symposium (International) on Combustion*, The Combustion Institute, pp. 1307–1313.

Chen, A. (1994). Application of Computational Fluid Dynamics to the Analysis of Inlet Port Design in Internal Combustion Engines, Ph.D. Thesis, Loughborough University.

Cotter, M. A. and Charles, M. E. (1993). Transient Cooling of Petroleum by Natural Convection in Cylindrical Storage Tanks – I. Development and Testing of a Numerical Simulator, *Int. J. Heat Mass Transfer*, Vol. 36, No. 8, pp. 2165–2174.

Durst, F. and Loy, T. (1985). Investigations of Laminar Flow in a Pipe with Sudden Contraction of Cross-sectional Area, *Comput. Fluids*, Vol. 13, No. 1, pp. 15–36.

Lockwood, F. C. and Malalasekera, W. M. G. (1988). Fire Computation: The Flashover Phenomenon, *Twenty-second Symposium (International) on Combustion*, The Combustion Institute, pp. 1319–1328.

Lockwood, F. C. and Shah, N. G. (1981). A New Radiation Solution Method for Incorporation in General Combustion Prediction Procedures, *Eighteenth Symposium (International) on Combustion*, The Combustion Institute, pp. 1405–1414.

Malalasekera, W. M. G. (1988). Mathematical Modelling of Fires and Related Processes, Ph.D. Thesis, Imperial College, London.

Schlichting, H. (1979). *Boundary-layer Theory*, 7th edn, McGraw-Hill, New York.

Index